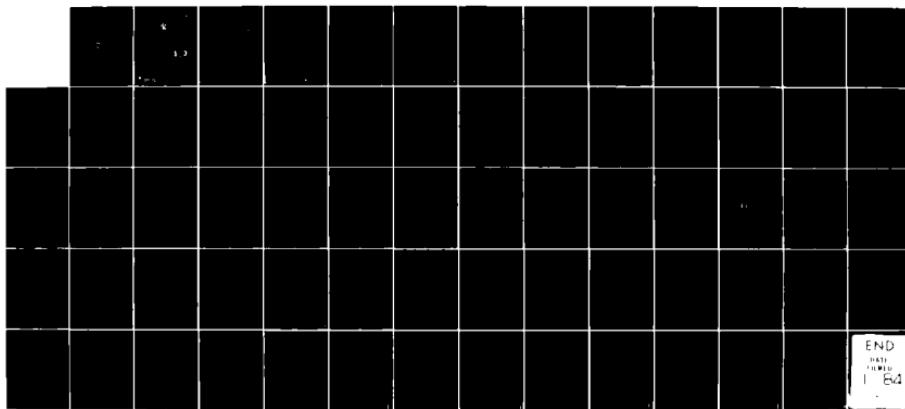


REF ID: A135 522      TIGS (TABLE PLOT INTERACTIVE GRAPHICS SYSTEM) - AN  
INTERACTIVE GRAPHICAL S... (U) NAVAL AIR DEVELOPMENT  
CENTER WARMINSTER PA AIRCRAFT AND CREW S... M J CADDY  
UNCLASSIFIED 01 MAY 83 NADC 83030 60

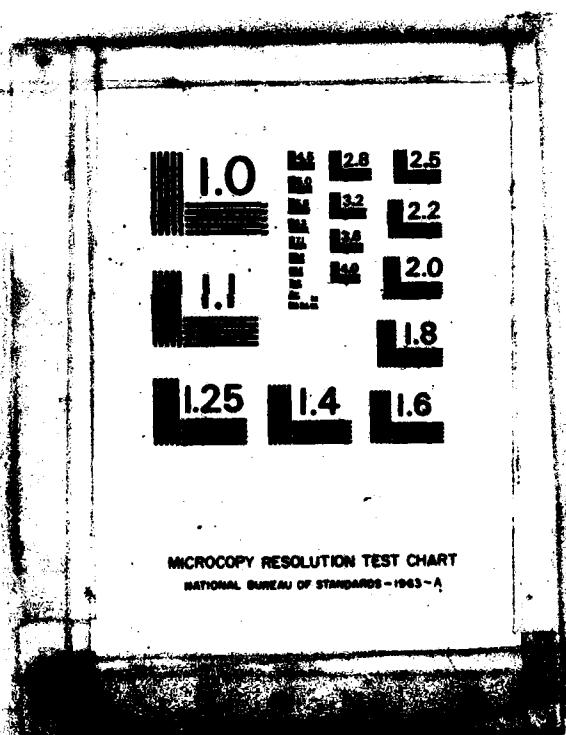
17

F/G 9/2

N1



END  
DATE  
TIME  
1-84



REPORT NO. NADC-63030-60

12



## TIGS-AN INTERACTIVE GRAPHICAL SYSTEM FOR THE CREATION AND CORRECTION OF TABULAR DATA SETS

AD-A1355522

M. J. Caddy  
Aircraft and Crew Systems Technology Directorate  
NAVAL AIR DEVELOPMENT CENTER  
Warminster, Pennsylvania 18974

1 MAY 1983

DTIC  
ELECTED  
S D  
DEC 7 1983

FINAL REPORT  
AIRTASK NO. A5365300/001F/8000SW050002  
Work Unit No. RP008  
AIRTASK NO. A2605100/001C/8W04770003-01  
Work Unit No. XN401/X05701

Prepared For  
NAVAL AIR SYSTEMS COMMAND  
Department of the Navy  
Washington, DC 20370

83 12 07 018

SECURITY CLASSIFICATION OF THIS PAGE (If other than Report)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NADC-83030-80	2. GOVT ACCESSION NO. <i>AD-A135 522</i>	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) TIGS - An Interactive Graphical System for the Creation and Correction of Tabular Data Sets	5. TYPE OF REPORT & PERIOD COVERED Final Report	
7. AUTHOR(N) M. J. Caddy	6. PERFORMING ORG. REPORT NUMBER	
9. PERFORMING ORGANIZATION NAME AND ADDRESS Aircraft & Crew Systems Technology Directorate (Code 8052) Naval Air Development Center, Warminster, Pa. 18901	10. PROGRAM ELEMENT, PROJECT, TASK, AREA & WORK UNIT NUMBERS A5365360700 1F/80596W0598002 RF808 & A2695100/001C/ 8W04770003-01 XN401/X570F	
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Air Systems Command Department of the Navy Washington, DC 20361	12. REPORT DATE 1 May 1983	
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	13. NUMBER OF PAGES 53	
	15. SECURITY CLASS. (of this report) UNCLASSIFIED	
	16. DECLASSIFICATION/DOWNGRADING SCHEDULE	
17. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Checklist on reverse side if necessary and identify by block number) Interactive Computer Graphics		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) A general purpose interactive graphical computer code is described which permits interactive graphical creation and correction of tabular data sets. This code was developed for Tektronics 4015 hardware utilizing the NADC 6800/Cyber 176 computer facilities.		

DD FORM 1 JAN 73 EDITION OF 1 NOV 68 IS OBSOLETE  
5100-000-0001

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (If other than Report)

## TABLE OF CONTENTS

	<u>Page</u>
<b>INTRODUCTION .....</b>	<b>3</b>
<b>DISCUSSION .....</b>	<b>3</b>
<b>CODE DEVELOPMENT .....</b>	<b>3</b>
<b>HARDWARE REQUIREMENTS .....</b>	<b>3</b>
<b>SOFTWARE OVERVIEW .....</b>	<b>5</b>
<b>USER EXPERIENCE .....</b>	<b>5</b>
<b>CONCLUSIONS .....</b>	<b>8</b>
<b>REFERENCES.....</b>	<b>8</b>
<b>APPENDIX A – USER'S GUIDE.....</b>	<b>A-1</b>
<b>INPUT CONSIDERATIONS .....</b>	<b>A-2</b>
<b>TABLE DATA FORMAT.....</b>	<b>A-2</b>
<b>EXAMPLES .....</b>	<b>A-2</b>
<b>LIMITATIONS .....</b>	<b>A-2</b>
<b>INTERACTIVE PROMPTING .....</b>	<b>A-2</b>
<b>TABLET INITIATION PROCEDURE .....</b>	<b>A-10</b>
<b>TIGS INTERACTIVE COMMANDS .....</b>	<b>A-12</b>
<b>EXAMPLE INTERACTIVE SESSION .....</b>	<b>A-13</b>
<b>APPENDIX B – TIGS CODE FORTRAN LISTING .....</b>	<b>B-1</b>

(1)

Accession For	
NTIS GRA&I <input checked="" type="checkbox"/>	
DTIC TAB <input type="checkbox"/>	
Unannounced <input type="checkbox"/>	
Justification _____	
By _____	
Distribution/ _____	
Availability Codes _____	
Distr	Avail and/or Special
(1)	

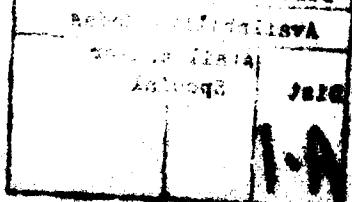
NADC-83030-60

LIST OF FIGURES

<u>Figure No.</u>		<u>Page</u>
1	Input and Output Axis Representation .....	4
2	TIGS Example Plot .....	6
3	TIGS Segmentation Modules .....	7
A-1	One Parameter Table Look-up .....	A-4
A-2	Two Parameter Table Look-up .....	A-6
A-3	Three Parameter Table Look-up .....	A-8
A-4	Tablet Command Menu .....	A-11
A-5	Example Plot .....	A-14
A-6	Example Plot .....	A-16
A-7	Example Plot .....	A-17
A-8	Example Plot .....	A-18
A-9	Example Plot .....	A-19
A-10	Example Plot .....	A-20
A-11	Example Plot .....	A-21
A-12	Example Plot .....	A-22
A-13	Example Plot .....	A-23
A-14	Example Plot .....	A-24
A-15	Example Plot .....	A-26
A-16	Example Plot .....	A-27
A-17	Example Plot .....	A-28

LIST OF TABLES

<u>Table No.</u>		<u>Page</u>
A-I	Data Input Instructions .....	A-3
A-II	One Parameter Card Inputs .....	A-5
A-III	Two Parameter Card Inputs .....	A-7
A-IV	Three Parameter Card Inputs .....	A-9



## INTRODUCTION

The NAVAIRDEVCEN (Naval Air Development Center) is a large user of vehicle and propulsion design and performance computer codes and is constantly seeking to improve their efficiency and flexibility. A significant number of these codes are dependent on the use of input tabular data sets. Quite frequently these data sets are initially received in a format incompatible with direct use in these codes, resulting in a time consuming, error prone transformation task. To circumvent this problem, development of a rapid data transformation code was undertaken. The impetus for this effort was the need to prepare for a planned substantial increase in analyses of various aircraft and propulsion systems.

This present report describes a code based on the use of an interactive graphics system that permits direct creation of digital tabular data sets from material in graph form, utilizing a Tektronics 4015 graphics terminal, digitizer tablet and hardcopy unit. In addition the user may edit and correct these data directly from the digitizer tablet or from the graphics display screen using cursor cross hairs and tablet commands. This code, entitled TIGS (Table Plot Interactive Graphics System) was developed using the NAVAIRDEVCEN CDC 6600/Cyber 175 computer facilities. A user's guide for this code is shown in Appendix A. A Fortran listing of the TIGS code is shown in Appendix B.

## DISCUSSION

### CODE DEVELOPMENT

The TIGS code was developed as a general purpose computer tool to permit the user to prepare and edit tabular data sets, using interactive graphics, prior to use in other computer codes. The tabular data sets may represent a functional relationship between a dependent variable and several independent variables, an example of which is shown in Figure 1. In this figure FXYZ is the dependent variable and is a function of the independent variables X, Y, and Z. The basic output of the code is graphical plots on a Tektronics 4015 type of storage tube graphics terminal along with a computer file consisting of the digital tabular data representation of that plot. These digital tabular data are suitable for use in nearly all of the vehicle and propulsion design computer codes used within the Aircraft and Crew Systems Technology Directorate at the NAVAIR-DEVCEN. Further details of the tabular data output format are discussed in the user's guide Appendix A and in reference (a). While the TIGS code is a stand-alone interactive system, the graphical executive portion of the code may be used in conjunction with any other user written code. In effect this flexibility permits the user to interactively prepare and edit data which in turn is passed to the user's code. Experience in using TIGS has shown that the time required to prepare data for use in the vehicle and propulsion design codes has been reduced by a factor of 10.

### HARDWARE REQUIREMENTS

The TIGS code is specialized in that it was written for a CDC 6600/Cyber 175 computer system using a 1200 baud line under the CDC telex time sharing system. Graphical implementations are provided by a Tektronics model 4015 terminal with the enhanced graphics option. A large Tektronics tablet may be employed in the digitization process along with a model 4631 hardcopy unit. The TIGS system could be modified for use with other graphics systems. Figure 2 shows a typical TIGS hardcopy plot.

NADC-89030-80

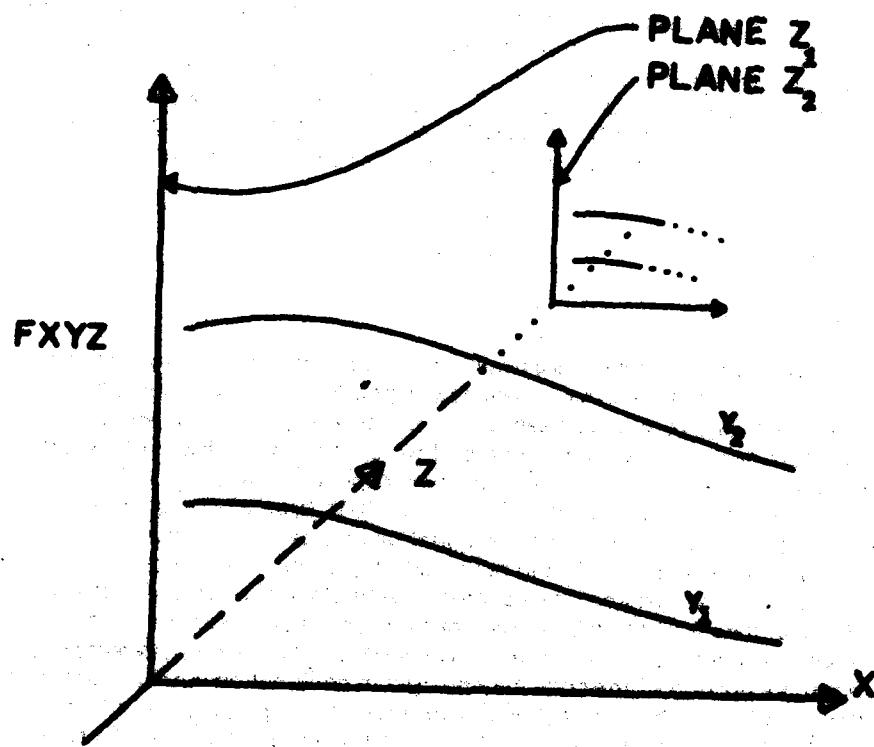


FIGURE 1. INPUT AND OUTPUT AXIS REPRESENTATION

## SOFTWARE OVERVIEW

The TIGS code is comprised of seven basic modules using the standard utility Tektronics release 3.2 software compiled under Fortran IV. TIGS uses the Cyber segmentation loader requiring about 40000 octal memory locations to execute. The segmentation setup consists of seven modules described below. The information flow between these modules is represented by Figure 3.

TIGS is the main executive module that controls the input and output and interplays with the graphics executive.

TABR contains the code to input and output the digital data in the required format.

TIGPPR is the graphics executive module. This module controls the graphical input and permits the user to interact with the graphical screen and digital tablet controlling data point values, plot sizes, curve options, titles and scaling.

The TIGPPR module performs these functions through connections to other segmentation modules GETVAL, LOPTIM, LABEL, and DRAWIT.

GETVAL is used to input data points either from the graphics screen or the digitizer tablet.

LOPTIM implements the axes scaling and grid options selected by the user.

LABEL uses the data values to compute the axes tic marks and other data related to fitting the plot on the graphical screen.

DRAWIT processes the scaling, axes, along with other plot data and generates the commands that draw the vectors on the graphic screen.

There are two basic operating modes in the graphics executive: creation and correction. In the creation mode a digital data file is created using the cross hair cursor either directly from the Tektronics screen or from the digitizer tablet. Commands from the screen are implemented by first positioning the cross hairs and then keying a single letter indicating the command. Commands from the digitizer tablet are implemented in two steps: first the command code letter is keyed using a tablet command menu; second, the coordinate position going with the command is keyed at the desired position. From either the screen or the tablet, the graphics executive receives the command and coordinate position. The commands received by the graphics executive are generally used in three different ways:

- 1) add, delete or change a coordinate point
- 2) change a graphics executive switch from off to on or on to off
- 3) control the size and view of the graphical plot.

Some commands available on the screen can not be used on the digitizer tablet. A more detailed discussion of these commands is found in the user's guide, Appendix A.

## USER EXPERIENCE

The TIGS system has proved to be a very powerful, flexible tool. Task cost reductions of 10 to 1 have been shown to date using TIGS to prepare tabular inputs for other codes.

NADC-83030-60

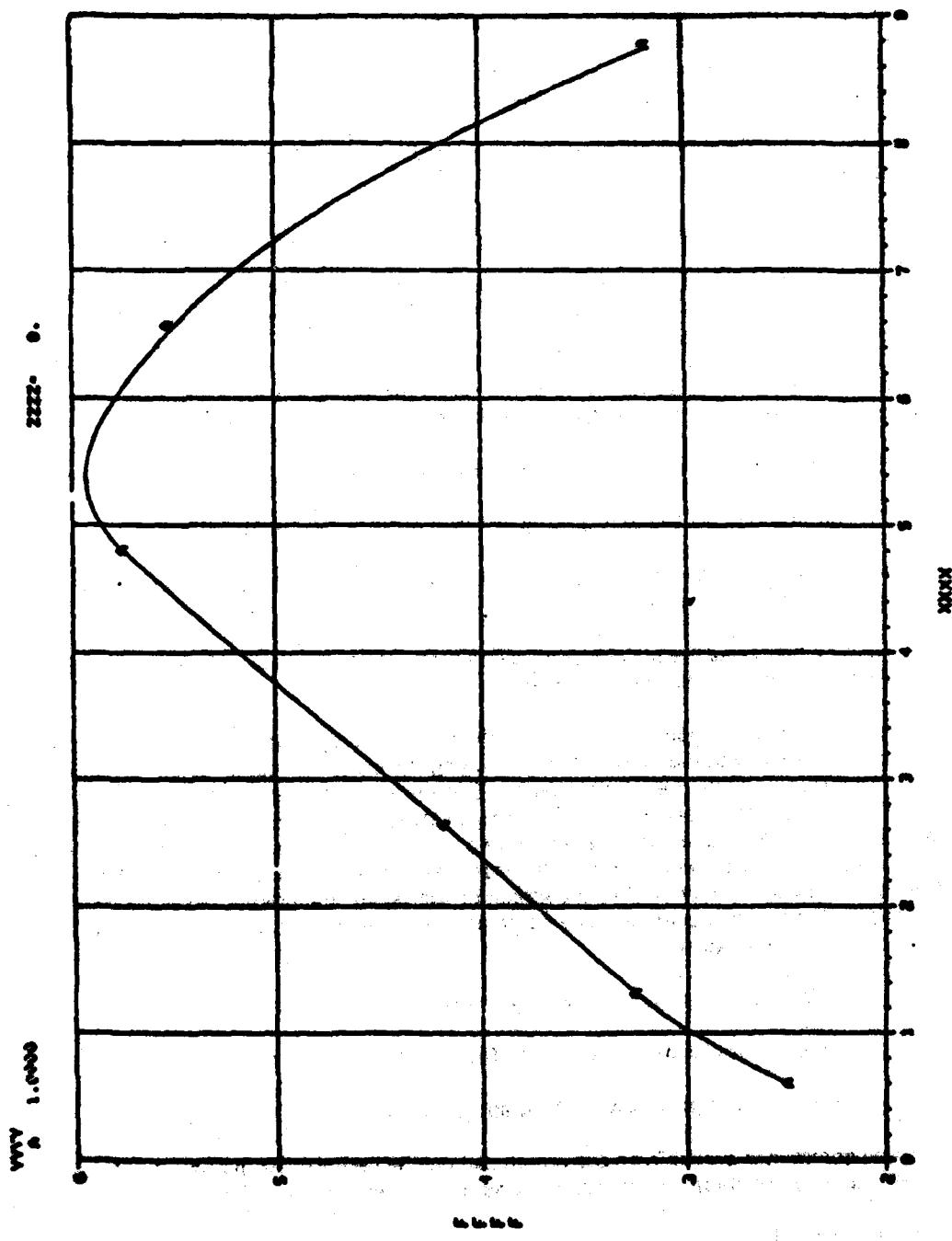


FIGURE 2. TIGB EXAMPLE PLOT

NADC-83030-60

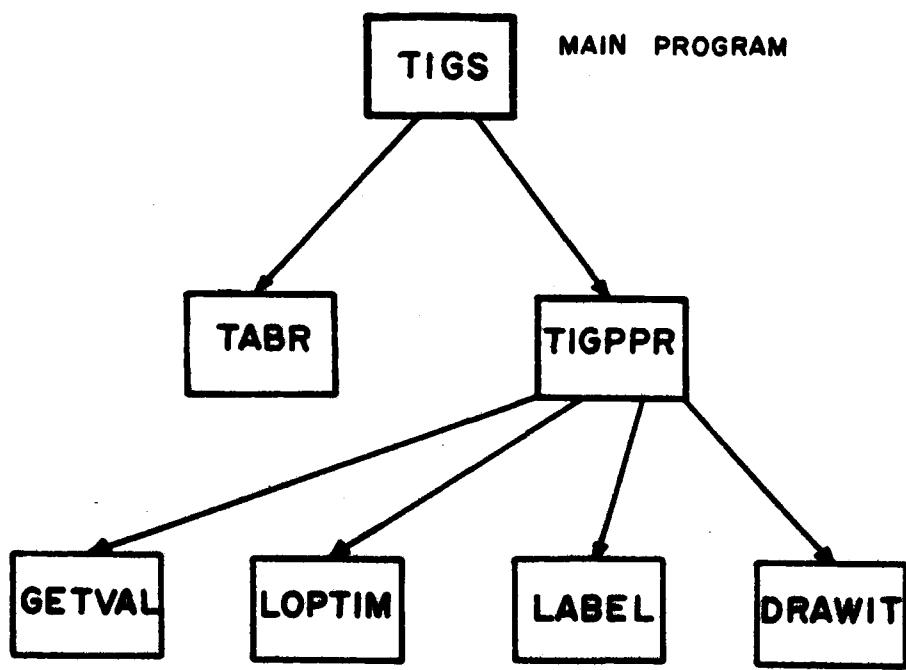


FIGURE 3. TIGS SEGMENTATION MODULES

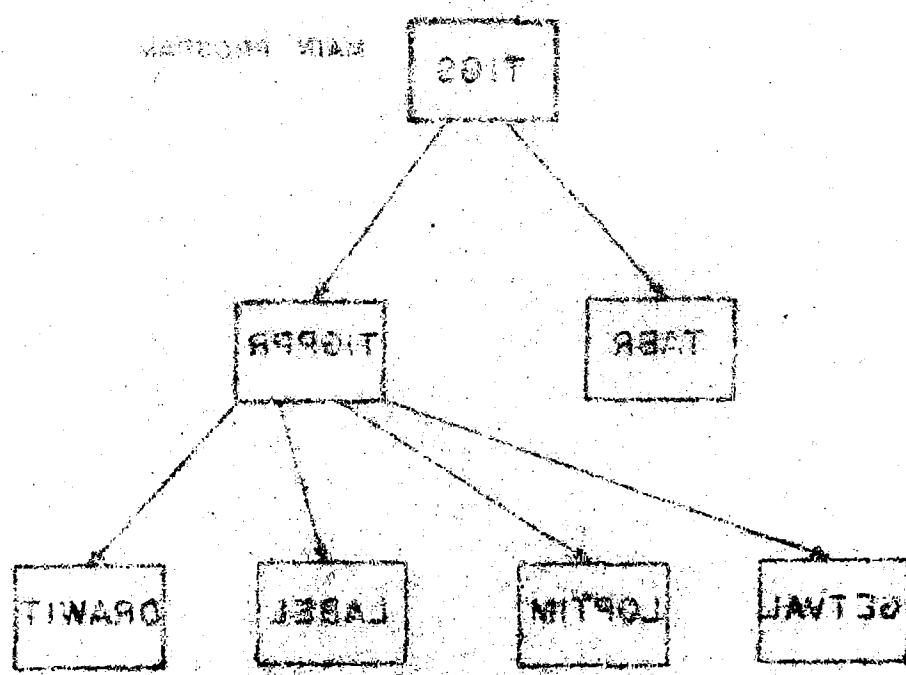
## ~~NADC REPORT~~

### CONCLUSIONS

An interactive graphical code system has been developed which is capable of rapid transformation of graphical information into tabular data formats which are compatible with the input requirements for a large variety of in-house programs.

### REFERENCES

- (a) Cuddy Michael J., "TREAD/TLOOK - Multipurpose Computer Routine for Interpolation and Extrapolation of Tabular Data" NADC Report 76366-30, 1977



**NADC-83030-60**

**APPENDIX A  
USER'S GUIDE**

## A.1 INPUT CONSIDERATIONS

The TIGS code will permit a direct creation of a data file from screen and or tablet commands. In addition, existing table data, input as file TAPE1, may be edited and corrected. In either case a new table data source with corrections is produced as an output on the TAPE7 file. The format of files TAPE1 and TAPE7 is the same. In the next section this format is illustrated.

## A.2 TABLE DATA FORMAT

The table data may represent a dependent (output) parameter as a function of 1, 2, or 3 independent (input) parameters. The basic method for inputting these tables is described in reference (a). However, for the sake of completeness, the user's guide portion of reference (a) has been extracted and duplicated herein and includes those modifications introduced since its initial publication. This information is shown in Table A-1.

## A.3 EXAMPLES

Card input data set-ups for three different examples are illustrated as follows:

### Example 1 (Drag coefficient as a function of Mach number)

The dependent variable is drag coefficient and the independent variable is Mach number. Figure A-1 illustrates the graphical relationship. This is a one parameter table look-up so the other two parameters are dummies. Table A-1 shows the card set-up for this example. The EOT (end of table) parameter label terminates the data for this table.

### Example 2 (Drag coefficient as a function of Mach number and lift coefficient)

The dependent variable is drag coefficient and the independent variables are Mach number and lift coefficient, illustrated in Figure A-2. This is a two parameter table look-up so that the third parameter is a dummy. Table A-2 shows the card set-up. In Table A-2 the last Mach parameter data repeats the previous Mach parameter data. In this situation, the last Mach parameter data card can be omitted. As a general rule, whenever the data on the X parameter axis as shown in Figure A-1, is repeated, then the X parameter data card need not be repeated.

### Example 3 (Drag coefficient as a function of Mach number, lift coefficient and CG location)

The dependent variable is drag coefficient and independent variables are Mach number, lift coefficient, and CG location, illustrated in Figure A-3. Table A-3 shows the card set-up for this three parameter example. Note that the input card set-up is symmetrical in that each CL parameter data card begins data for each CG parameter.

## A.4 LIMITATIONS

The TIGS system as presently written is limited to a maximum of 30 curves per plot, 150 points per curve, or a total of 300 points per plot. For example, a plot with 10 curves could be described with 5 curves using 40 points per curve, and the remaining 5 curves using 20 points per curve.

## A.5 INTERACTIVE PROMPTING

The TIGS code has been designed to prompt the user in supplying information in the correct format. Selection of the baud rates compatible with available transmission lines is possible. After

**TABLE A-I**  
**DATA INPUT INSTRUCTIONS**

Card No.		Format										
1	Table reference number of table look-up function; table title or descriptive information	1X,I4,7A10										
2	4 character identifier (user selected) used to identify the third independent variable. If table look-up has 2 or less independent variables, use a dummy identifier; the number of values of the third independent variable (must be less than 100.); values of the third independent variable arranged in ascending order.	A4,I3,3X,7F10.0										
2a,b,etc.	Continuation of third independent variable array, if required	10X,7F10.0										
3 and following	<p>All remaining cards have the same format as card 2, 2a, b, etc. The item which distinguishes the card types is the value of the independent variable. The 4 character identifiers of each independent variable must not be identical. The 4 characters of each independent variable card (after the title card) are user selected. The card order of each independent variable is significant. The first four cards with respective independent variables are as follows:</p> <table border="1" data-bbox="394 952 985 1296"> <thead> <tr> <th data-bbox="394 952 657 984"><u>Card</u></th><th data-bbox="739 952 870 984"><u>Definition</u></th></tr> </thead> <tbody> <tr> <td data-bbox="394 1005 657 1058">2,2a,b,etc.</td><td data-bbox="739 1005 952 1058">third independent variable, identifier and values</td></tr> <tr> <td data-bbox="394 1079 657 1132">3,3a,b,etc.</td><td data-bbox="739 1079 968 1132">second independent variable, identifier and values</td></tr> <tr> <td data-bbox="394 1153 657 1205">4,4a,b,etc.</td><td data-bbox="739 1153 935 1205">first independent variable, identifier and values</td></tr> <tr> <td data-bbox="394 1227 657 1279">5,5a,b,etc.</td><td data-bbox="739 1227 968 1279">dependent variable, identifier and values</td></tr> </tbody> </table> <p>The remaining input cards use these same identifier values as input above. On cards 4, 4a, b, etc. and 5, 5a, b, etc. are the dependent and first independent variable values along the line given by the first value of the second independent variable and in the plane of the first value of the third independent variable.</p> <p>Cards with the same respective identifier value cards 4, 4a, b, etc. and 5, 5a, b, etc. are repeated for different values of second independent variable until all second independent variables have been exhausted. The next card has an identifier corresponds to the second independent variable and new values of that variable for the plane of the second value of third independent variable. The values of the first independent variable need not be repeated if they are the same along each line of constant second independent variable. In each instance where the values are changed a new card is required.</p>	<u>Card</u>	<u>Definition</u>	2,2a,b,etc.	third independent variable, identifier and values	3,3a,b,etc.	second independent variable, identifier and values	4,4a,b,etc.	first independent variable, identifier and values	5,5a,b,etc.	dependent variable, identifier and values	
<u>Card</u>	<u>Definition</u>											
2,2a,b,etc.	third independent variable, identifier and values											
3,3a,b,etc.	second independent variable, identifier and values											
4,4a,b,etc.	first independent variable, identifier and values											
5,5a,b,etc.	dependent variable, identifier and values											
Last	Table input termination indicator, EOT	A8										
	All remaining tables for this input section follow the same pattern as above. To end the table read-in mode, a blank table reference number is input behind the last table of the entire table set.											

NADC-83030-60

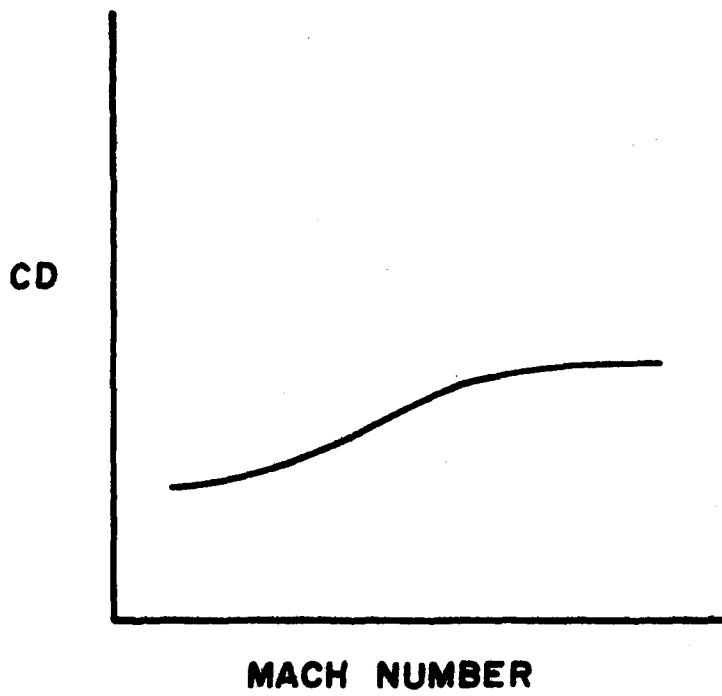


FIGURE A-1. ONE PARAMETER TABLE LOOK-UP

NADC-83030-60

TABLE A-II. ONE PARAMETER CARD INPUTS

◀ COLUMN LOCATION

123456789012345678901234567890123456789012345678901234567890

101	DRAG	COEFFICIENT VS MACH NUMBER		
Z 1	0.0			
Y 1	0.0			
MACH 4	0.0	0.1	0.2	0.3
CD 4	0.010	0.011	0.0112	0.0115
EOT				

NADC-83030-60

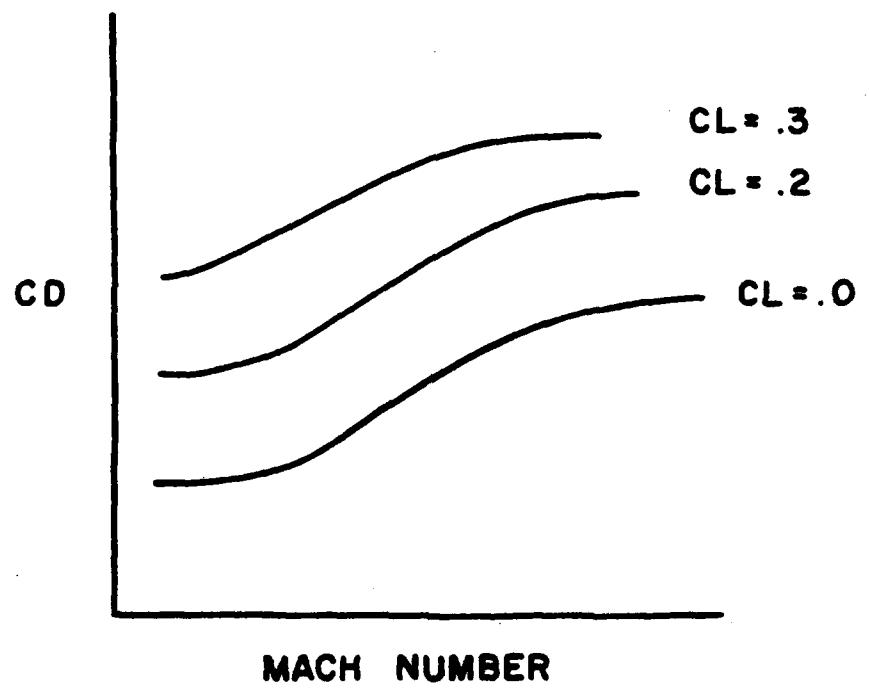


FIGURE A-2. TWO PARAMETER TABLE LOOK-UP.

NADC-83030-60

TABLE A-III. TWO PARAMETER CARD INPUTS

COLUMN LOCATION

1234567890123456789012345678901234567890123456789012345678901234567890

104	DRAG	COEFFICIENT VS M AND CL		
Z	1	0.0		
CL	3	0.0	0.2	0.3
MACH	4	0.0	0.1	0.2
CD	4	0.01	0.02	0.03
MACH	3	0.0	0.15	0.2
CD	3	0.1	0.02	0.03
MACH	3	0.0	0.15	0.2
CD	3	0.02	0.03	0.04
EOT				

NADC-83030-60

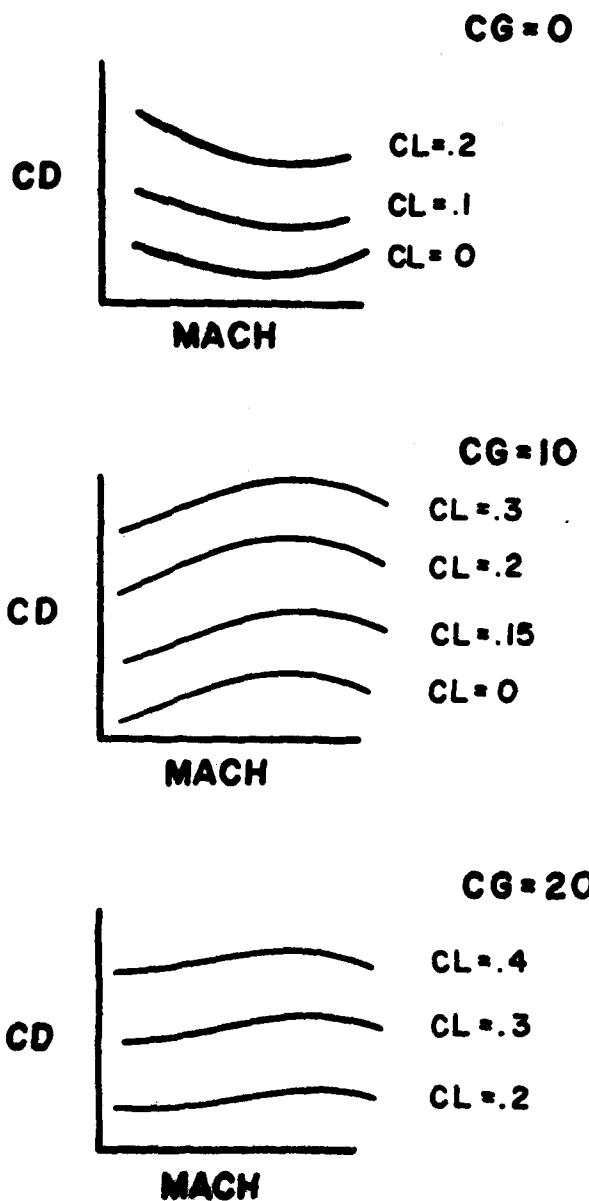


FIGURE A-3. THREE PARAMETER TABLE LOOK-UP

NADC-83030-60

TABLE A-IV. THREE PARAMETER CARD INPUTS

← COLUMN LOCATION

1234567890123456789012345678901234567890123456789012345678901234567890

226 CD VS M, CL, AND CG			
CG	3	0.0	10.0
CL	3	0.0	0.1
MACH	4	0.0	0.2
CD	4	0.01	0.02
CD	4	0.02	0.03
MACH	3	0.0	0.1
CD	3	0.03	0.031
CL	4	0.0	0.15
MACH	3	0.0	0.2
CD	3	0.011	0.011
CD	3	0.015	0.015
CD	3	0.020	0.020
CD	3	0.025	0.025
CL	3	0.2	0.3
MACH	4	0.0	0.2
CD	4	0.01	0.01
MACH	3	0.0	0.2
CD	3	0.011	0.011
CD	3	0.021	0.022
EOT			

logging into the host system and the baud rate has been selected, different prompts will appear depending on the user response to the initial interactive query. The response will depend on whether the user intends to correct an existing file or create a new file via the screen or tablet. After the baud rate selection the next query to appear will be:

(a) "IS THIS A CREATION RUN?"

A "Y" response indicates a TAPE1 file is to be newly created and the following prompts will appear.

(b) "ENTER TABLE TITLE CARD  
(COLUMNS 1-5 SHOULD BE TABLE REFERENCE NUMBER)"

The user should refer to the instructions in section A-2 Table A-1, card 1.

(c) "ENTER 4 CHARACTERS FOR EACH LABEL FOR Z, Y, X, FXYZ  
(separated by commas)"

The user should refer to instructions in section A-2 Table A-1, card 2.

(d) "ENTER NUMBER OF Z VALUES"

The user should refer to instructions in section A-2 Table A-1, card 2.

(e) "ENTER Z VALUES IN ASCENDING ORDER"

The user should now enter the values of the Z parameter with blanks or commas between the data pieces.

(f) "WANT TO SPECIFY DECIMAL PLACES ON TAPE2?"

An "N" response will by default, set the number of places at the maximum allowable. If the user enters a "Y", this will be prompt query (g).

(g) "ENTER NUMBER OF DECIMAL PLACES FOR Z,Y,X,FXYZ"

The user should specify the number of decimal places (up to 9) separated by blanks or commas for Z,Y,X,FXYZ parameters.

At this point, the following message will appear:

"NO DATA TO BE FOUND...ENTER COMMAND"

The user may now begin creating the tabular data set with either a "N" (new line) command or a "T" (tablet operation) command. The reader is referred to sections A-5 and A-6 for additional information.

If the response to the initial query (query (a) above) is "N" then this means that data on the TAPE1 is to be used and queries (b) thru (e) are skipped.

#### A.6 TABLET INITIATION PROCEDURE

Tablet commands are issued using a command menu. The command menu is a section of the tablet, 20 one inch squares (10 columns by 2 rows) in which keying the coordinates within a

square is interpreted as the indicated command. Figure A-4 shows the positions of the menu commands within the 10 by 2 inch squares.

Tablet operation begins initially by attaching the command menu at any convenient location on the tablet. The menu should be approximately parallel with the lower edge of the tablet. Upon first entering the "T" command, the user enters the position of the upper left corner of the menu.

The coordinates of this position are then used to determine the commands. Next, the user attaches the graph that is to be used at a convenient tablet location. Squaring the graph is not necessary since any angular correction required is performed in TIGS. The user then may select any convenient orthogonal axes and enters the following requested information:

- a) the coordinate position of the crossing point of the orthogonal axes, and the coordinate values X and FXYZ respectively at the crossing point.
- b) the coordinate position of any X axis point and its value. (usually this coordinate position is the maximum axis length)

After these entries have been made the "NO DATA FOUND TO PLOT" message will appear. At this time the user may issue commands from the tablet menu.

#### A.7 TIGS INTERACTIVE COMMANDS

Commands from the screen involve only positioning the cross hairs and keying the appropriate command. Commands from the tablet involve first selecting the command from the menu and then indicating the coordinate position. Once a tablet command has been set it remains set until changed. The user is free to change to and from tablet and screen command modes. The following commands are available:

- "A"- add point after. The user positions the cross hairs and keys the "A" command (or indicates the tablet command and position). The system will respond by drawing the symbol at the new point. (Note. See the "C" command for further discussion.)
- "B"- add point before. This command is exactly like the "A" command except that the point is added before the pointer position.
- "C"- position the pointer to the array location that the user wishes to add a new point. The next command following the "C" command to add a point may be an "A" to add after or a "B" to add before the pointer position. In addition an "M" command may be used to move to a new location the point indicated by the position pointer. A "V" command may also be used. It should be noted that the pointer position after each added point becomes the position of the added point. Possible valid commands would be "CAABBAVVVA" permitting the user to continuously add new points very rapidly. Any other command drops the pointer position, which must be restored by another "C" command to add new points.
- "D"- delete the point closest to the cross hairs or pen position.
- "E"- end or terminate this plot and return to TIGS for next plot if any.
- "F"- format or change type of curve drawn for each line as follows:

NADC-83030-60

A	B	C	D	E	G	H	I	N	P
add	add	position	delete	end	grid	halt	initial	new	plot
after	before	pointer	point	plot	switch	tablet	tablet	line	data
R	S	V	W						
restore	show	value	window						
window	value	input	data						

FIGURE A-4. TABLET COMMAND MENU

ITIP- switch determining type of curve drawn (ITIP=2, default) ITIP can have the following values: (Note: a negative value will have the same meaning except no symbols are drawn.)

- 0 indicates symbols only, no curve drawn
- 1 indicates linear fit
- 2 indicates smooth spline like fit with respect to x axis
- 3 same as ITIP=2 except with respect to y axis
- 4 indicates data is multivalued and the fit is with respect to arc length along curve
- 5 indicates data is multivalued and forms a closed figure; the fit is with respect to arc length and joined at the ends.

NOTE: if data is not in ascending order when ITIP = 2 or ITIP = 3, then curve fit will default to ITIP = 4; DEFAULT format is ITIP = 2.

- "G"- IGRID switch- turns grid from on to off or off to on.
- "H"- halt tablet and returns control to screen. This only applies to tablet modes.
- "I"- initialize tablet starting with graph coordinate locations. This only applies to tablet mode.
- "M"- move the point indicated by the pointer position to the new coordinates indicated by the cross hairs.
- "N"- begin a new curve at point indicated. Whenever this command is issued, the user will also enter the new curve value. The pointer position becomes the new point permitting commands such as "NAAMAAAABBBVAB".
- "P"- re-plot data with scale as shown. see Note
- "R"- rescale data to the largest size and re-plot. see Note
- "S"- show the current coordinate values at the position indicated.
- "V"- values input; same as "A" or "B" command except the actual X and F coordinate values are input.
- "W"- window the plot to fit within the rectangle indicated by the diagonal between two coordinate positions (two points are sent).

Note: The commands "F", "P" and "R" have a dual meaning. If the vertical cross hair is to the left of the vertical plot axis then:

"F" indicates to change the ITIP of the curve indicated by the position of the horizontal cross hairs. "P" indicates activate TIGS to plot the curve indicated by the position of the horizontal cross hairs. This action is cumulative in that one or more of many curves may be indicated in this manner. "R" is used to deactivate this special mode and thus restore all curves to plotting status.

### A.8 EXAMPLE INTERACTIVE SESSION

In this example the TIGS system is used to create a plot. The figures in this example are actual copies of what the user would see on the Tektronix screen. In these figures a "?" followed by data indicates that these data were the user's response to the indicated query. In the following discussion numbers enclosed by circles refer to corresponding numbers on a figure pointing to a feature under discussion. In figure A-5 (1) is the command used to begin execution of TIGS.

(2) is the user response to the query as to transmission line rate. (3) indicates that a creation is requested and results in queries (4) thru (7). (4) is the main title on the plot preceded by the table reference number. The table reference number should be a 5 digit integer number. The title can be up to 4 lines. The user may indicate a new line by leaving three consecutive blanks between words. (5) is the response to the query requesting four variable names for the respective data. Note that each variable name must be 4 characters in length; blanks count as characters. (6) is the response to the number of Z variables requested. Each Z value represents a single plane. (7) is the response to input each Z value. (8) is the response to the decimal place query related to the TAPE7 file. This file is an output file containing all of the data generated during this session. Each prompt, as shown, indicates that the data on TAPE7 will contain the maximum decimal places that will fit with each space. Some caution is necessary if the user specifies the number of decimal places for each parameter; precision could be lost if a low number of decimal places is initially selected. A good technique is to examine the TAPE7 file with the maximum decimal places specified first and then re-enter TIGS, if necessary, and specify decimal places as required.

When the user responds to (8), figure A-6 will be displayed. The meaning of figure A-6 is that a plot command was implied and that data was not found to plot. This is a proper response since the user, through (3), on figure A-5, elected a creation run and there is no data as yet to plot. The user will notice for the first time that cross hairs also have appeared on the screen. (Note these are not shown in figure A-6). The cross hair is a prompt signal that an input is requested. The input is a single upper case letter. A "RETURN" is not required after typing the single letter command. The single letter command issued in this example was a "N" indicating a new line. The response to this command shown in figure A-7. The first prompt, (9), requests one set of data coordinates, X and FXYZ, for one point. The purpose of this is to scale the final plot. The response at (10) is a value assigned to this line (this one set of coordinates is the beginning of a potential curve).

After entering the number one for this query as noted by (10), the screen will appear as shown in figure A-8. The "Y" shown on the left top of this figure is the 4 character label entered in (5) figure A-5. The "A" and number under the "Y" is the symbol for the first line and the line value assigned to it as, (10) in figure 7. The "A" in the center of the plot at (0,0) is the first point (and only point) of line A. The pointer positioned message indicates that the reference point from which to add points has been identified. This occurred automatically since only one point at this time is in the plot, the first point. All of the other 4 character labels, including the main plot label, are also shown. The value of the plot plane (Z value) is zero and is shown at the top right.

In the next steps the user has moved the cross hairs and "keyed" the "A" characters indicating "add point after". The "add point after" in this context means that the data point storage of the new point is after the point indicated by pointer position. The curves are always drawn in the order towards the "after" point. After each point is added the pointer position becomes the position of the added point. Figure A-9 shows the addition of added four points as they would appear on the screen. Figure A-10 is a reprint of the data resulting from the user keying a "P". This command simply plots a curve through the data points shown.

Figure A-11 is a resize and reprint resulting from an "R" command. The plot axes have been rescaled to permit the largest plot of the data points that will fit within the screen.

NADC-83030-60

T1BS ← ①  
ENTER BAUD RATE CODE  
1=1200, 2=2400, 3=4800, 4=9600 ← ②  
? 4 ← ②  
T1BS VER 2.0 11-14-80 ← ③  
IF THIS IS A CREATION RUN ENTER Y ← ③  
? Y ← ③  
ENTER TABLE TITLE CARD ← ④  
(COLUMNS 1-5 SHOULD BE THE TABLE REFERENCE NUMBER)  
? 00001 TEST EXAMPLE CREATION ← ④  
ENTER 4 CHARACTERS FOR EACH LABEL FOR Z,Y,X,FXYZ ← ⑤  
(SEPARATED BY COMMAS) ← ⑤  
? Z ,Y ,X ,FXYZ ← ⑤  
ENTER NUMBER OF Z VARIABLES--- FREE FORM ← ⑥  
? 1 ← ⑥  
ENTER Z VALUES ,ASCENDING ORDER--FREE FORM ← ⑦  
? 6 ← ⑦  
WANT TO SPECIFY DECIMAL PLACES ON TAPE?? ← ⑧  
? N ← ⑧

FIGURE A-8. EXAMPLE PLOT

**NADC-83030-60**

**NO DATA FOUND TO PLOT ..ENTER COMMAND**

**FIGURE A-8. EXAMPLE PLOT**

NADC-83030-60

NO DATA FOUND TO PLOT ..ENTER COMMAND  
Input X,Y  
? 0 0 ← ⑨  
Input Line Value  
? 1 ← ⑩

FIGURE A-7. EXAMPLE PLOT

NADC-83030-60

1.0000  
..Polar Positionned

TEST EXAMPLE CREATION

$$z = 0.$$

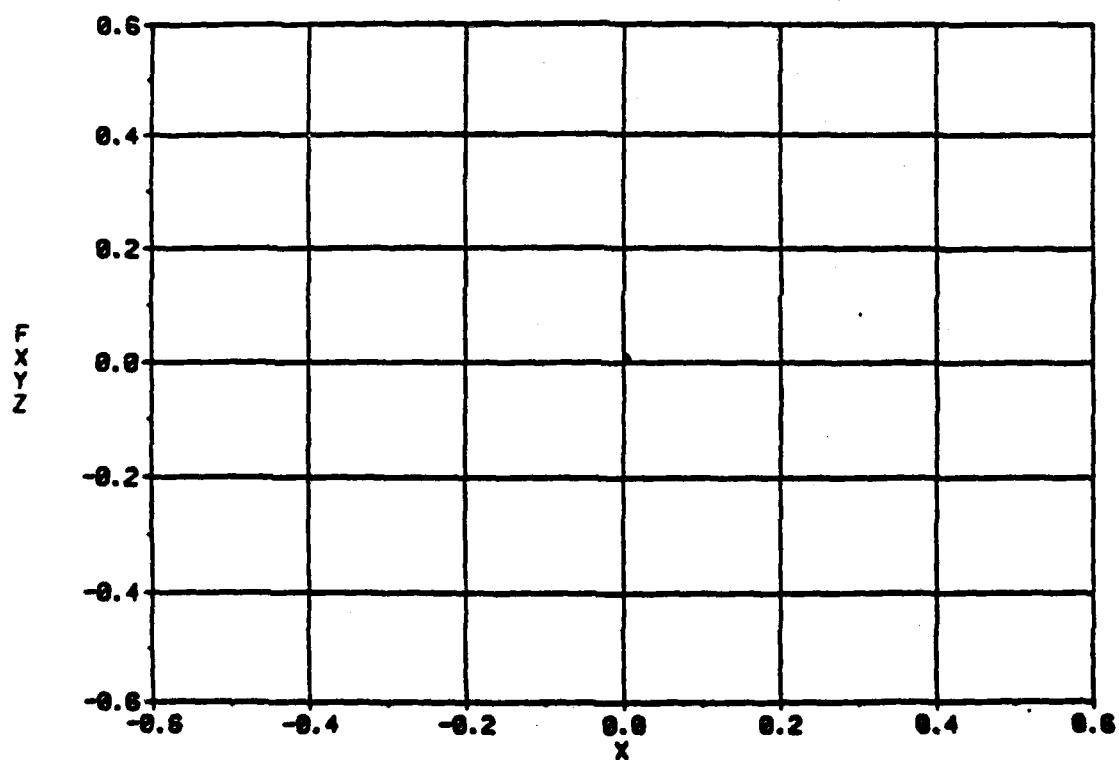


FIGURE A-8. EXAMPLE PLOT

NADC-83030-60

A 1.000  
..Polarer Positions

TEST EXAMPLE CREATION

$z = 0.$

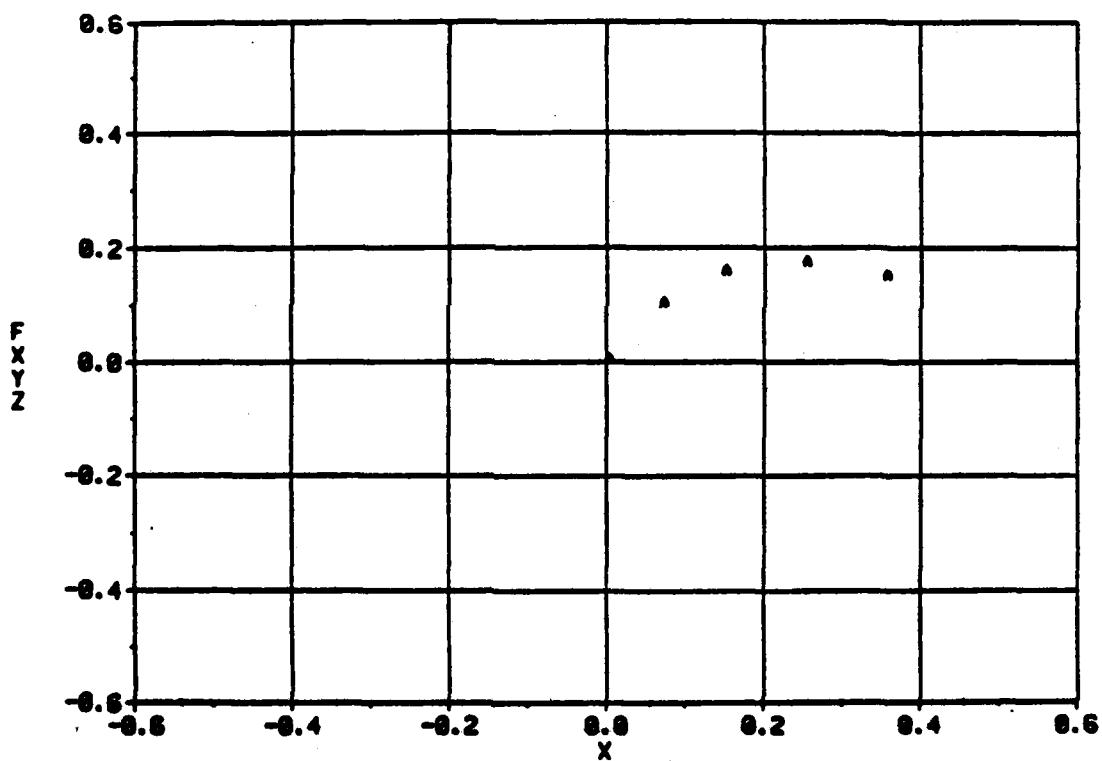


FIGURE A-9. EXAMPLE PLOT

NADC-83030-60

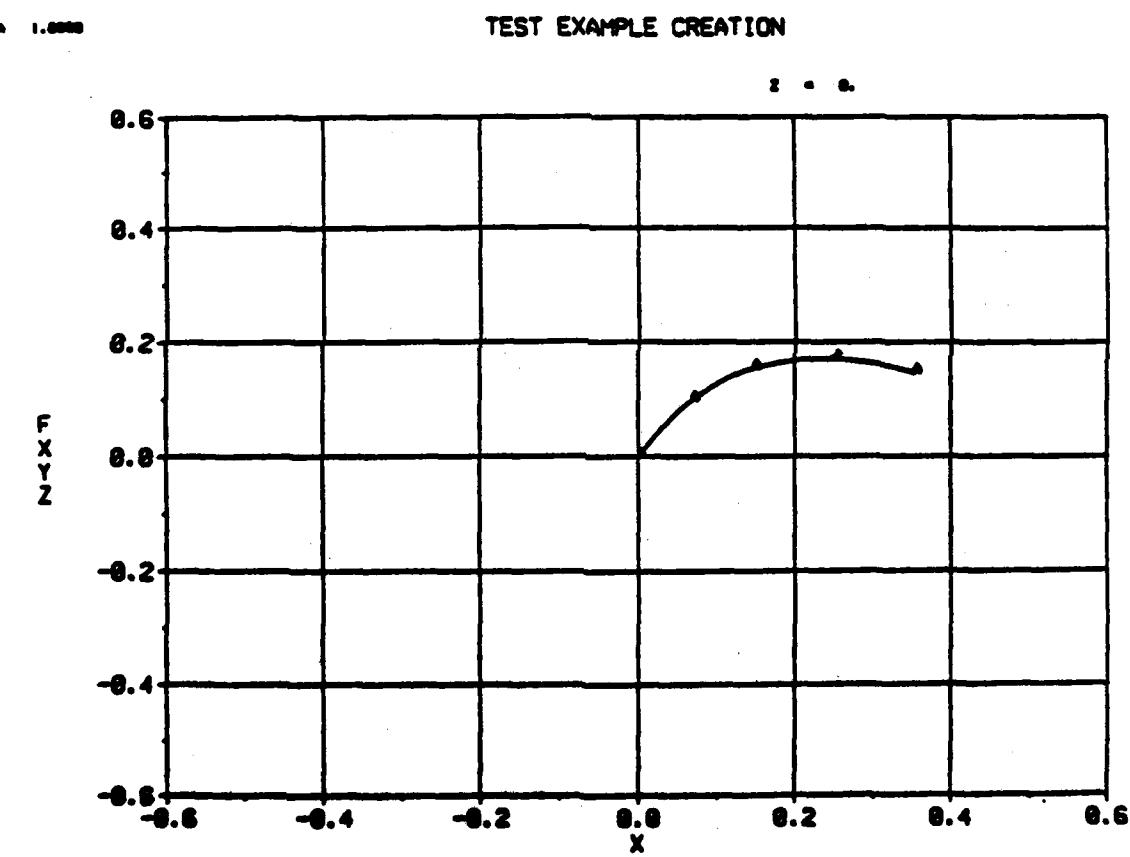


FIGURE A-10. EXAMPLE PLOT

NADC-83030-60

TEST EXAMPLE CREATION

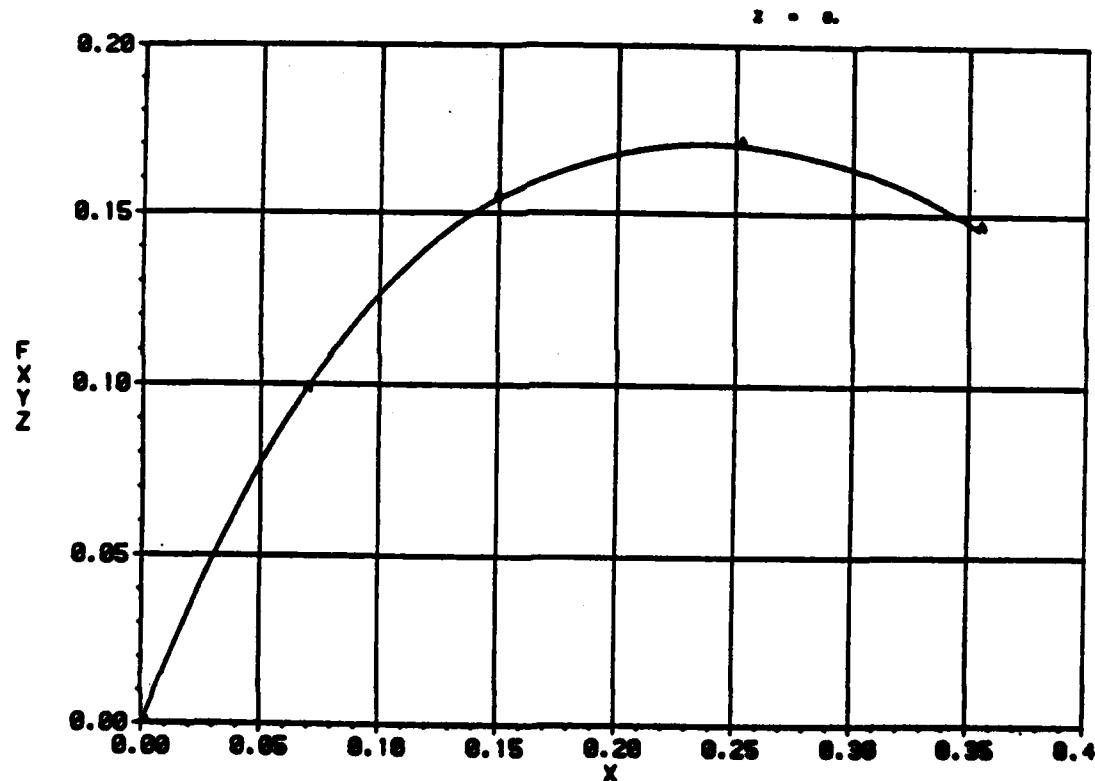


FIGURE A-11. EXAMPLE PLOT

NADC-83030-60

In figure A-12 the cursor was first positioned at approximate co-ordinate locations of .15 and .16 and a "C" command issued. The "C" indicates to identify the closest point to the intersection of the cross hairs as pointer position. This command also resulted in the message stating "pointer positioned" at the top left. The four points shown on figure 12 were then added by the user moving the cursor and "Keying" the "A" command. In figure A-13 a "P" command was issued first and then the cross hairs were at the position indicated by the "B" symbol and the "N" command was keyed. This resulted in the "input line value" query shown in the top left of figure A-13. In figure A-14 the query response is shown and the user has inputted more points by just moving the cursor and using the "A" command.

In figure A-15 the user has replotted the data with a "P" command and then the cursor was positioned near the end "B" point at X=.14 and a "C" command was keyed. The next command sent by the user was a "V", to input an exact value. This prompted the query "Input X,Y" to appear. The last query "A or B mode?" simply request that the user identify where in the data storage is the new data point stored, before the pointer or after the pointer.

Figure A-16 is a final plot of the data showing the new point. At this point the user keyed an "E" command and "ended" the execution. In figure A-17, the output file created during this example is listed using the CED text editor, showing all the data points.

NADC-83030-60

V 1.0000  
..Polar Coordinates

TEST EXAMPLE CREATION

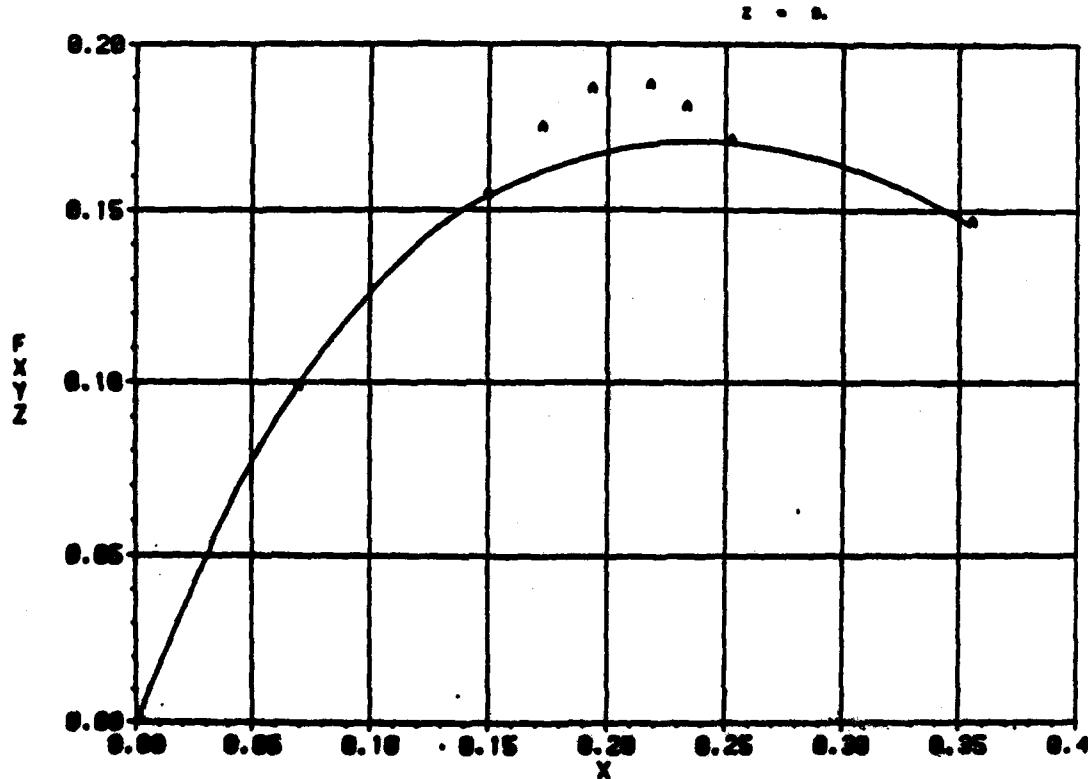


FIGURE A-12. EXAMPLE PLOT

NADC-83030-60

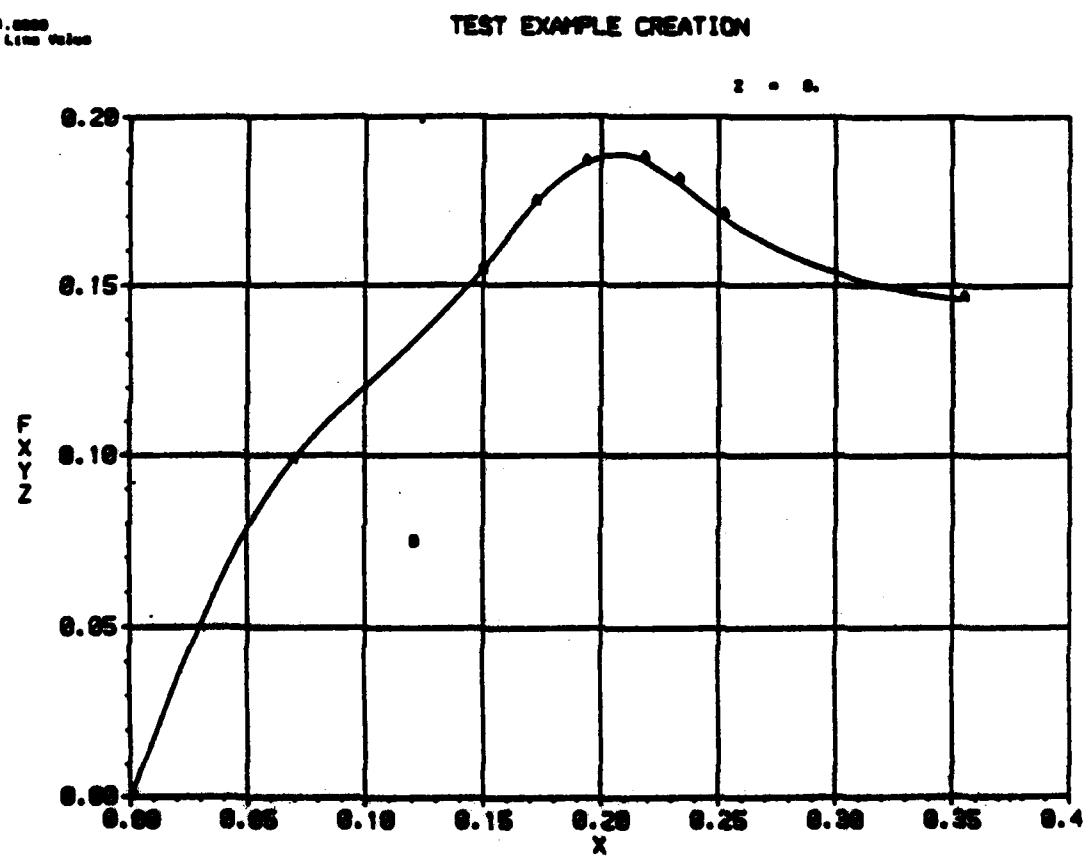


FIGURE A-13. EXAMPLE PLOT

NADC-83030-60

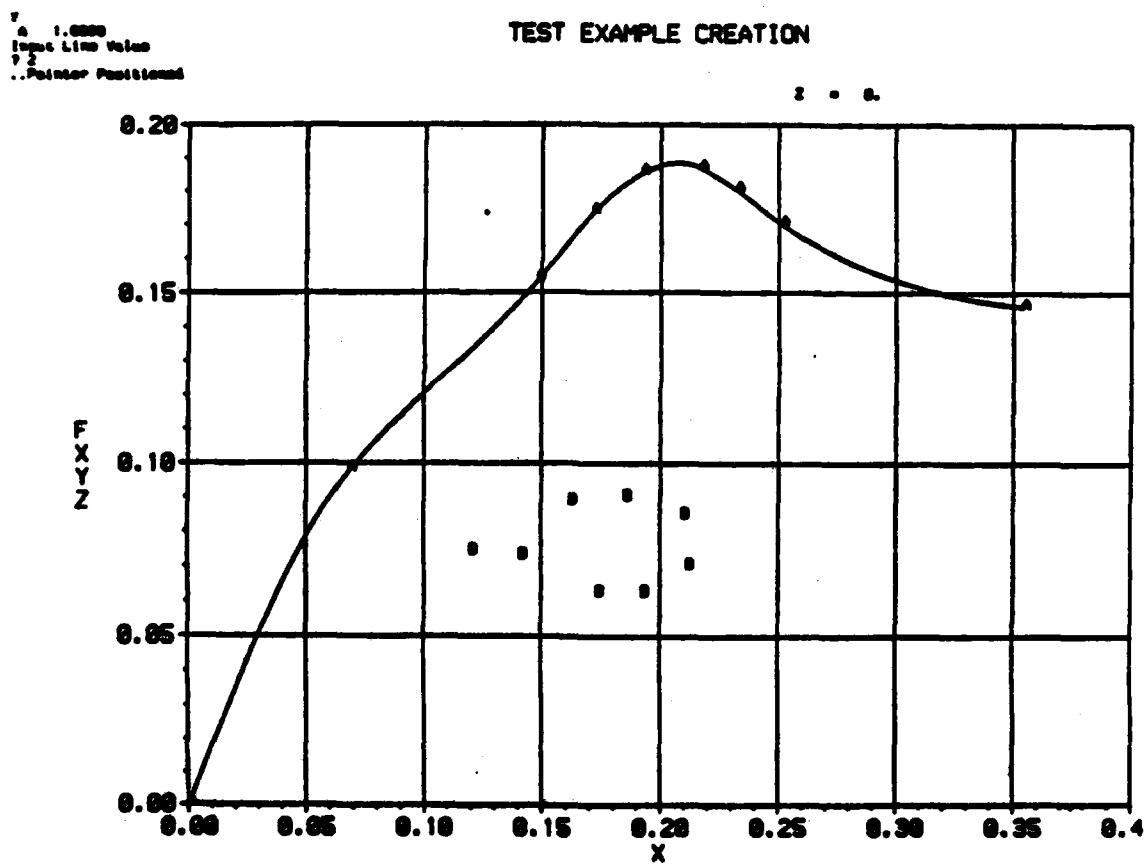


FIGURE A-14. EXAMPLE PLOT

NADC-83030-60

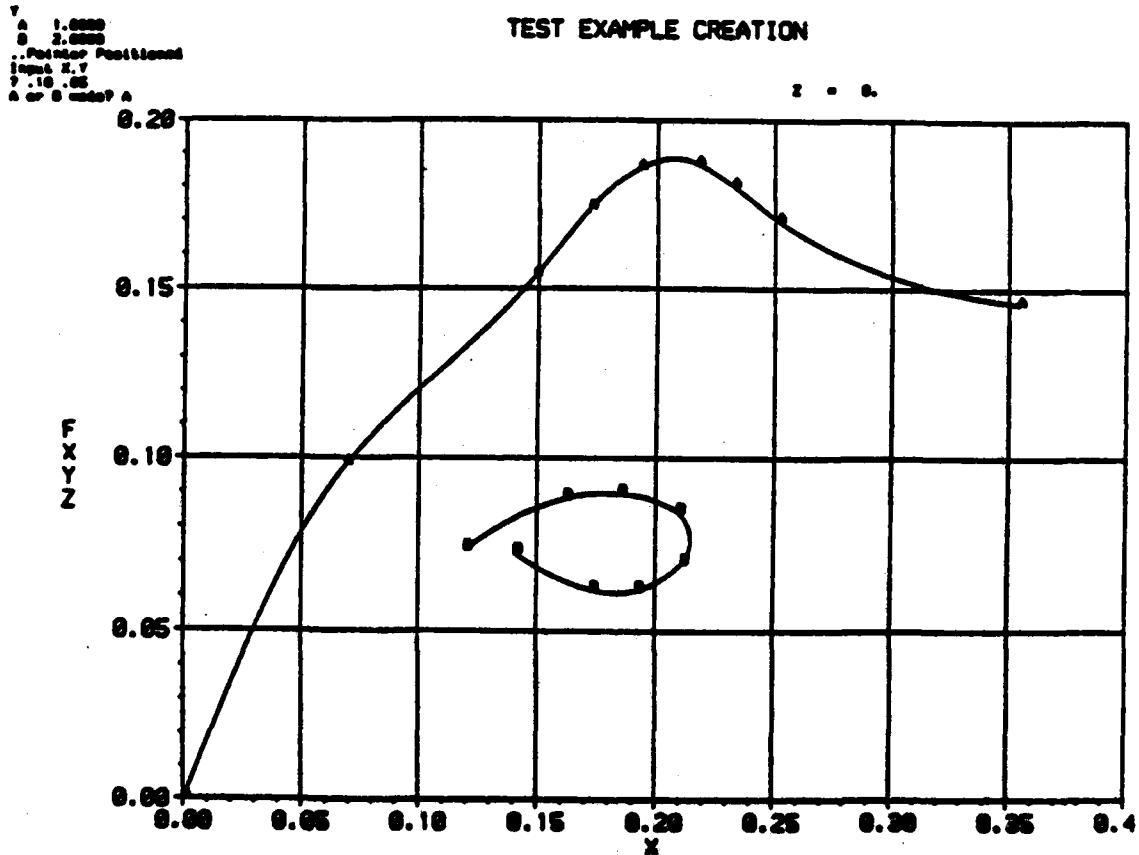


FIGURE A-15. EXAMPLE PLOT

NADC-83030-60

1:1000

TEST EXAMPLE CREATION

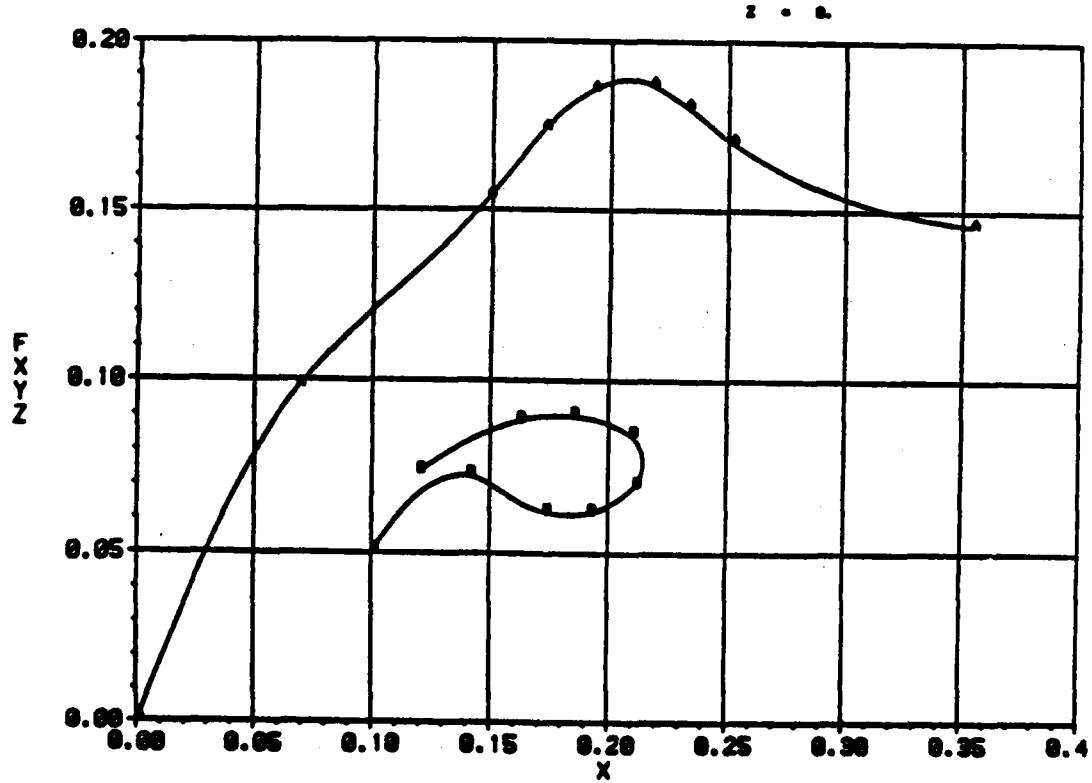


FIGURE A-10. EXAMPLE PLOT

NADC-83030-60

END T105  
.737 CP SECONDS EXECUTION TIME  
/CED,TAPE7  
CED 1.2  
\? Pa  
00001 TEST EXAMPLE CREATION  
Z 1 0.  
Y 2 1.0000000 2.0000000  
X 0 0.0000000 .0005714 .1405714 .1714286 .1623818 .2171429 .2323818  
.2514286 .2642857  
FXYZ 0 0.0000000 .0000000 .1540000 .1730007 .1650007 .1670000 .1690000  
.1700000 .1460000  
X 0 .1200000 .1610040 .1847610 .2005230 .2114286 .1623818 .1730000  
.1400024 .1600000  
FXYZ 0 .8730007 .8632332 .8600007 .8642330 .8600007 .8610007 .8610007  
.8723330 .8600000  
EOT  
--END--  
--EOF--  
END OF INFORMATION  
\? S  
ABORTED  
/

FIGURE A-17. EXAMPLE PLOT

**NADC-83030-60**

**APPENDIX B  
FORTRAN LISTING**

```

CTIGS          TIGS0002
PROGRAM TIGS(INPUT=101,COUTPUT,TAPE1=101,TAPE7=101,TAPE5=INPUT) TIGS0003
C *****
C*****          TIGS TPLOT INTERACTIVE GRAPHICS SYSTEM TIGS0004
C*****          M CADDY JAN 30 78 TIGS0005
C*****          DIMENSION LT(7),XV(30),NPTS(30),X(300),Y(300),Z(30),A(99) TIGS0006
C*****          DATA NPLOT/0/ TIGS0007
C*****          DATA NT/8/ TIGS0008
C*****          DATA ITIP,IGRID/2,1 / TIGS0009
C*****          DATA NPTS,XV/30*0,30*0./ TIGS0010
C*****          DATA IEND/10HEOT / TIGS0011
10 FORMAT(A5,7A10) TIGS0012
REWIND 1 TIGS0013
REWIND 7 TIGS0014
PRINT 20 TIGS0015
20 FORMAT(* TIGS VER 2.0 8/2/78 *, TIGS0016
1      /* IF THIS IS A CREATION RUN ENTER Y*) TIGS0017
READ 30,IC TIGS0018
30 FORMAT(1R1)
IC=IC-30B
IF(IC.EQ.1) GO TO 40
C*****
C*****          FILE IS NOT BEING CREATED READ IT FROM TAPE1 TIGS0019
C*****          31 READ (1,10) LNO,LT TIGS0020
IF(LNO.EQ.10H      ) GO TO 251 TIGS0021
C*****
Ca****          CALL IN Z VALUES TIGS0022
C*****
CALL TABR(LZ,NZ,Z,1) TIGS0023
TIGS0024
C*****
C*****          READ IN Y,X,FXYZ DATA FOR NON CREATION RUN TIGS0025
C*****          CALL TABR(LY,NY,A,1) TIGS0026
TIGS0027
CALL TABR(LX,N,X,1) TIGS0028
TIGS0029
CALL TABR(LF,N,Y,1) TIGS0030
TIGS0031
GO TO 100 TIGS0032
TIGS0033
40 PRINT 50 TIGS0034
TIGS0035
50 FORMAT(* ENTER TABLE TITLE CARD*/,
1 * (COLUMNS 1-5 SHOULD BE THE TABLE REFERENCE NUMBER*)*) TIGS0036
TIGS0037
READ 10,LNO,LT TIGS0038
TIGS0039
C*****
C*****          READ TABLE NUMBER AND TITLE TIGS0040
C*****          PRINT 60 TIGS0041
TIGS0042
60 FORMAT(* ENTER 4 CHARACTERS FOR EACH LABEL FOR Z,Y,X,FXYZ*)/ TIGS0043
TIGS0044
TIGS0045
TIGS0046
TIGS0047

```

```

1 * (SEPARATED BY COMMAS)*) TIGS0048
C**** TIGS0049
C**** READ TITLES FOR EACH VARIABLE 4 CHARACTERS LONG TIGS0050
C**** TIGS0051
      READ 70,LZ,LY,LX,LF TIGS0052
      70 FORMAT(4(A4,1X)) TIGS0053
C**** TIGS0054
C**** GET NUMBER OF Z VARIABLES AND VALUES TIGS0055
C**** TIGS0056
      PRINT 80,LZ TIGS0057
      80 FORMAT(* ENTER NUMBER OF *,A4,* VARIABLES--- FREE FORM*) TIGS0058
      CALL GETIN(1,Z) TIGS0059
      NZ=Z(1) TIGS0060
      PRINT 90,LZ TIGS0061
      90 FORMAT(* ENTER *,A4,* VALUES ,ASCENDING ORDER--FREE FORM*) TIGS0062
      CALL GETIN(NZ,Z) TIGS0063
C**** TIGS0064
C**** WRITE TO TAPE7 TITLE CARD AND TABLE NUMBER TIGS0065
C**** TIGS0066
      100 WRITE(7,10) LNO,LT TIGS0067
      PRINT 110,LZ,LY,LX,LF TIGS0068
      110 FORMAT(* ENTER NUMBER OF DECIMAL PLACES FOR *,4(A4,1X) TIGS0069
      1 ,* FREE FORM*) TIGS0070
C**** TIGS0071
C**** GET NUMBER OF DECIMAL PLACES FOR EACH VARIABLE TIGS0072
C**** TIGS0073
      CALL GETIN(4,XV) TIGS0074
      LZDP=XV(1) TIGS0075
      LYDP=XV(2) TIGS0076
      LXDP=XV(3) TIGS0077
      LFDP=XV(4) TIGS0078
C**** TIGS0079
C**** WRITE TO TAPE7 THE Z VALUES ETC... TIGS0080
C**** TIGS0081
      CALL TFORM(1,LZ,NZ,Z,LZDP,7) TIGS0082
C**** TIGS0083
C**** INITIALIZE TEK SOFTWARE TIGS0084
C**** TIGS0085
      CALL INITT(120) TIGS0086
      CALL TERM(3,4096) TIGS0087
      CALL CHRSIZ(4) TIGS0088
      DO 250 IZ=1,NZ TIGS0089
C**** TIGS0090
C**** IF CREATION MODE THEN SET DEFAULTS TO 0 TIGS0091
C**** TIGS0092
      IF(IC.NE.1) GO TO 120 TIGS0093
      NPTS(1)=0 TIGS0094
      X(1)=0. TIGS0095

```

NADC-83030-60

```

Y(1)=0.                                     TIGS0096
GO TO 210                                    TIGS0097
C****                                     TIGS0098
C****      NON CREATION MODE               TIGS0099
C****                                     TIGS0100
120 CONTINUE                                TIGS0101
      K=1                                     TIGS0102
C****                                     TIGS0103
C****      TRANSFER SECOND INDEPENDENT VARIABLE TO XV ARRAY TIGS0104
C****                                     TIGS0105
      DO 130 J=1,NY                           TIGS0106
130 XV(J)=A(J)                               TIGS0107
      IF(IZ.EQ.1) GO TO 140                  TIGS0108
      CALL TABR(LX,N,X,1)                     TIGS0109
      CALL TABR(LF,N,Y,1)                     TIGS0110
140 LNX=N                                    TIGS0111
      LNY=N                                    TIGS0112
      NPTS(1)=N                                TIGS0113
      NPTS(2)=0                                TIGS0114
C****                                     TIGS0115
C****      READ NEXT SET                   TIGS0116
C****                                     TIGS0117
150 CALL TABR(LW,N,A,1)                      TIGS0118
C****                                     TIGS0119
C****      CHECK FOR NEXT Z GROUP          TIGS0120
C****                                     TIGS0121
      IF(LW.EQ.LY) GO TO 210                 TIGS0122
C****                                     TIGS0123
C****      CHECK FOR END OF TABLE          TIGS0124
C****                                     TIGS0125
      IF(LW.EQ.4HEOT ) GO TO 210            TIGS0126
C****                                     TIGS0127
C****      CHECK FOR NEXT X DATA           TIGS0128
C****                                     TIGS0129
      IF(LW.NE.LX) GO TO 170                 TIGS0130
C****                                     TIGS0131
C****      DATA IS X DATA STORE IT        TIGS0132
C****                                     TIGS0133
      LOX=LNX                                TIGS0134
      DO 160 J=1,N                           TIGS0135
      LNX=LNX+1                             TIGS0136
160 X(LNX)=A(J)                            TIGS0137
      GO TO 150                             TIGS0138
C****                                     TIGS0139
C****      DATA HAD BETTER BE LY          TIGS0140
C****                                     TIGS0141
      170 IF(LW.NE.LF) STOP                TIGS0142
C****                                     TIGS0143

```

C****	IF DATA HAS NOT BE INPUT FOR X DATA USE LAST VALUES	TIGS0144
C****		TIGS0145
	IF(LNX.GT.LNY) GO TO 190	TIGS0146
	LL=LOX	TIGS0147
	DO 180 J=1,N	TIGS0148
	LNX=LNX+1	TIGS0149
	LL=LL+1	TIGS0150
	180 X(LNX)=X(LL)	TIGS0151
C****		TIGS0152
C****	UPDATE COUNTERS	TIGS0153
C****		TIGS0154
	190 K=K+1	TIGS0155
	NPTS(K)=N	TIGS0156
	NPTS(K+1)=0	TIGS0157
C****		TIGS0158
C****	LOAD Y DATA	TIGS0159
C****		TIGS0160
	DO 200 J=1,N	TIGS0161
	LNY=LNY+1	TIGS0162
	200 Y(LNY)=A(J)	TIGS0163
C****		TIGS0164
C****	GO BACK TO GET NEXT GROUP	TIGS0165
C****		TIGS0166
	GO TO 150	TIGS0167
C****		TIGS0168
C****	PLOT DATA	TIGS0169
C****		TIGS0170
	210 CALL TIGPPR(NPLOT,LF,1,LX,1,LT,8,X,Y,NPTS,LY,1,XV,LYDP,ITIP,	TIGS0171
1	IGRID,LZ,Z(IZ))	TIGS0172
	CALL ANMODE	TIGS0173
C****		TIGS0174
C****	COUNT NUMBER OF Y VALUES	TIGS0175
C****		TIGS0176
	NY=0	TIGS0177
	DO 220 I=1,30	TIGS0178
	IF(NPTS(I).EQ.0) GO TO 230	TIGS0179
	NY=NY+1	TIGS0180
	220 CONTINUE	TIGS0181
	GO TO 250	TIGS0182
C****		TIGS0183
C****	WRITE TO TAPE7 Y DATA ETC....	TIGS0184
C****		TIGS0185
	230 CALL TFORM(1,LY,NY,XV,LYDP,7)	TIGS0186
	LOC=1	TIGS0187
	J=0	TIGS0188
	240 J=J+1	TIGS0189
	NP=NPTS(J)	TIGS0190
	IF(NP.EQ.0) GO TO 250	TIGS0191

```

C**** TIGS0192
C**** WRITE TO TAPE7 X DATA ETC... TIGS0193
C**** TIGS0194
CALL TFORM(LOC,LX,NP,X,LXDP,7) TIGS0195
C**** TIGS0196
C**** WRITE TO TAPE7 Y DATA ETC... TIGS0197
C**** TIGS0198
CALL TFORM(LOC,LF,NP,Y,LFDP,7) TIGS0199
LOC=LOC+NP TIGS0200
GO TO 240 TIGS0201
250 CONTINUE TIGS0202
WRITE (7,10) IEND TIGS0203

C**** TIGS0204
C**** IF NON CREATION MODE THEN GO BACK TO READ NEXT TITLE TIGS0205
C**** TIGS0206
IF(IC.NE.1) GO TO 31 TIGS0207
251 WRITE (7,10) TIGS0208
REWIND 7 TIGS0209
END TIGS0210

CTFORM TFOR0001
SUBROUTINE TFORM(LOC,LAB,N,X,IP,K) TFOR0002
DIMENSION X(1),IFORM(3) TFOR0003
C**** FORMATTING SUBROUTINE FOR TPLOT FORMAT TFOR0004
C**** LOC IS THE LOCAL ARRAY POSITION TO PRINT FROM TFOR0005
C**** LAB IS THE 4 CHARACTER LABEL TFOR0006
C**** N IS THE NUMBER TO PRINT TFOR0007
C**** X IS THE ARRAY CONTAINING THE VALUES TFOR0008
C**** IP IS THE NUMBER OF DECIMAL PLACES TO USE IN FORMAT TFOR0009
IF(IP.LT.0) IP=0 TFOR0010
IF(IP.GT.9) IP=9 TFOR0011
JO=LOC-1 TFOR0012
NP=N TFOR0013
IF(NP.GT.7) NP=7 TFOR0014
IFORM(1)=10H(A4,I3,3X, TFOR0015
IFORM(2)=55555555420634335733B+IP TFOR0016
IFORM(3)=10H) TFOR0017
WRITE(K,IFORM) LAB,N,(X(I+JO),I=1,NP) TFOR0018
IFORM(1)=10H( TFOR0019
IF(N.GT.7) WRITE(K,IFORM)(X(I+JO),I=8,N) TFOR0020
RETURN TFOR0021
END TFOR0022

CTABR TABR0001
SUBROUTINE TABR(LAB,N,A,K) TABR0002
DIMENSION A(1) TABR0003
READ(K,10) LAB,N,(A(I),I=1,7) TABR0004
10 FORMAT(A4,I3,3X,7F10.0) TABR0005
IF(N.GT.7) READ(K,20) (A(I),I=8,N) TABR0006
20 FORMAT(10X,7F10.0) TABR0007

```

```

30 ISUB(I)=I+1 TIGP0047
C
C MERGE HERE TO REPLOT TIGP0048
C
40 CALL BINITT TIGP0049
LCNT=3120 TIGP0050
IGRID1=(3*IGRID+7)*.5 TIGP0051
C SUM UP NUMBER OF POINTS TIGP0052
NL=0
NPTOT=0
DO 60 I=1,30
N=NPTA(I)
IF(N.EQ.0) GO TO 70
NL=NL+1
60 NPTOT=NPTOT+N TIGP0061
C SET STORAGE LIMIT TO NPTOT FIRST PASS TIGP0062
70 IF(NSTOR.EQ.0) NSTOR=NPTOT TIGP0063
IF(NPTOT.GT.0)GO TO 90 TIGP0064
NSTOR=0 TIGP0065
CALL MOVABS(0,LCNT) TIGP0066
CALL ANMODE TIGP0067
PRINT 80 TIGP0068
80 FORMAT(* NO DATA FOUND TO PLOT ..ENTER COMMAND*) TIGP0069
LCNT=LCNT-LDEL TIGP0070
IPLOT=0 TIGP0071
GO TO 200 TIGP0072
C
C SECOND INDEPENDENT VARIABLE TITLE TIGP0073
C
90 IF(NCC.LE.0)GO TO 140 TIGP0074
CALL MOVABS(0,LCNT) TIGP0075
CALL ANMODE TIGP0076
PRINT 110,(LABVAL(J1),J1=1,NCC) TIGP0077
CALL MOVABS(2800,2800) TIGP0078
CALL ANMODE TIGP0079
PRINT 100,LZ,ZVAL TIGP0080
100 FORMAT(A4,*,G13.5) TIGP0081
110 FORMAT(8A10) TIGP0082
LCNT=LCNT-LDEL TIGP0083
KL=0 TIGP0084
KH=55B TIGP0085
DO 130 J1=1,NL TIGP0086
LCNT=LCNT-LDEL TIGP0087
CALL MOVABS(0,LCNT) TIGP0088
KL=KL+1 TIGP0089
CALL ANMODE TIGP0090
PRINT 120,KH,KL,VLABL(J1) TIGP0091
TIGP0092
TIGP0093
TIGP0094
TIGP0095
TIGP0096
TIGP0097
TIGP0098
TIGP0099

```

```

RETURN TABR0008
END TABR0009
CTIGP TIGP0001
C TIGP0002
C TEK INTERACTIVE GPPR M CADDY FEB 78 TIGP0003
C TIGP0004
SUBROUTINE TIGPPR(NPLOT,LABY,N1,LABX,N2,LABTL,NT ,X,Y, TIGP0005
1 NPTA,LABVAL,NCC,VLBL,NDECVIN,ITIP,IGRID,LZ,ZVAL) TIGP0006
COMMON/TKTRNX/ITEKC(60) TIGP0007
DIMENSION X(200),Y(200),LABTL(9),NPTA(30),VLABL(30), TIGP0008
1 LABX(5),LABY(5),VTEM(8),LABVAL(8),IQUICK(30),ISUB(300) TIGP0009
DIMENSION MSG1(20),MSG2(20),MSG4(10),MSG5(10),MSG6(15),IALTM(6,2) TIGP0010
EQUIVALENCE (BEG(1),XBEG),(BEG(2),YBEG) TIGP0011
EQUIVALENCE (DEL(1),DELX),(DEL(2),DELY),(ITAB,LTV(2)) TIGP0012
EQUIVALENCE (EN(1),XEND),(EN(2),YEND) TIGP0013
EQUIVALENCE (IOFF,ITEKC(30)),(TXMIN,ITEKC(1)) TIGP0014
SET LINE SPACING TIGP0015
COMMON/TEKGPPR/LDEL,LCNT,MAXSR,LTV(17),EN(2),DEL(2),BEG(2),RDX2, TIGP0016
1 RDY2,NLINE,NDRAW(30),MODE(30) TIGP0017
DATA MSG1/46,q46,80,111,105,110,116,101,114,32, TIGP0018
1 80,111,115,105,116,105,111,110,101,100/ TIGP0019
DATA MSG2/73,110,112,117,116,32,76,105,110,101, TIGP0020
1 32,86,97,108,117,101,32,32,32,32/ TIGP0021
C**** ILLEGAL MESSAGE TIGP0022
DATA MSG4/42,73,108,108,101,103,97,108,42,32/ TIGP0023
DATA MSG5/73,110,112,117,116,32,88,44,89,32/ TIGP0024
DATA MSG6/65,32,111,114,32,66,32,109,111,100, TIGP0025
1 101,63,32,32,32/ TIGP0026
DATA ((IALTM(I,J),I=1,6),J=1,2)/65,102,116,101,114,32, TIGP0027
1 66,101,102,111,114,101/ TIGP0028
DATA IQUICK/0,0,1,1,2,3,10,4,0,0, TIGP0029
1 0,0,0,5,0,6,0,7,8,0, TIGP0030
2 0,0,9,0,0,0,0,0,0,0,0/ TIGP0031
A=1./KIN(1.) TIGP0032
LDEL=50 TIGP0033
IF(NPLOT.GT.0) GO TO 20 TIGP0034
DO 10 I=1,8 TIGP0035
10 LTV(I)=0 TIGP0036
20 IWIN=0 TIGP0037
NPLOT=NPLOT+1 TIGP0038
IGRID=1
DO 21 I=1,30
21 MODE(I)=ITIP
NTL=NT
NLINE=0
NSTOR=0
SET STORAGE POINTER TO INITIAL SEQUENCE
DO 30 I=1,299

```

120	FORMAT(1X,2R1,G13.5)	TIGP0100
130	CONTINUE	TIGP0101
C		TIGP0102
C	PREPARE TEKTRONIX AGII COMMON	TIGP0103
C		TIGP0104
140	CONTINUE	TIGP0105
	IPILOT=IPLOT+1	
	CALL CHRSIZ(4)	TIGP0106
C	SET SCREEN WINDOW SIZE	TIGP0107
	CALL SLIMX(640,4000)	TIGP0108
	CALL SLIMY(300,2700)	TIGP0109
C	SET TICK SIZES	TIGP0110
C	CALL XTICS(14)	TIGP0111
C	CALL YTICS(10)	TIGP0112
	IF(IWIN.NE.0) GO TO 170	TIGP0113
	AXMAX=-1.E99	TIGP0114
	AYMAX=-1.E99	TIGP0115
	AXMIN=+1.E99	TIGP0116
	AYMIN=+1.E99	TIGP0117
C	SET MIN AND MAX DATA VALUES	TIGP0118
	K=1	TIGP0119
	DO 150 I=1,NPTOT	TIGP0120
	AXMIN=AMIN1(AXMIN,X(K))	TIGP0121
	AYMIN=AMIN1(AYMIN,Y(K))	TIGP0122
	AXMAX=AMAX1(AXMAX,X(K))	TIGP0123
	AYMAX=AMAX1(AYMAX,Y(K))	TIGP0124
	KLAST=K	TIGP0125
C	SET KLAST TO END STORAGE VALUE	TIGP0126
150	K=ISUB(K)	TIGP0127
	IWIN=1	TIGP0128
	IF(AXMIN.NE.AXMAX) GO TO 160	TIGP0129
	AXMIN=AXMIN-.5	TIGP0130
	AXMAX=AXMAX+.5	TIGP0131
160	IF(AYMIN.NE.AYMAX) GO TO 170	TIGP0132
	AYMIN=AYMIN-.5	TIGP0133
	AYMAX=AYMAX+.5	TIGP0134
C	SET VIRTUAL WINDOW	TIGP0135
170	CALL DLIMX(AXMIN,AXMAX)	TIGP0136
	CALL DLIMY(AYMIN,AYMAX)	TIGP0137
	CALL XLEN(28)	TIGP0138
	CALL YLEN(28)	TIGP0139
	CALL XFRM(IGRID1)	TIGP0140
	CALL YFRM(IGRID1)	TIGP0141
	NBASE=IBASEX(0)	TIGP0142
	DO 180 I=1,2	TIGP0143
	CALL LOPTIM(NBASE)	TIGP0144
	CALL WIDTH(NBASE)	TIGP0145
	CALL SPREAD(NBASE)	TIGP0146

CALL TSET(NBASE)	TIGP0147
180 NBASE=IBASEY(0)	TIGP0148
EN(1)=COMGET(IBASEX(27))	TIGP0149
EN(2)=COMGET(IBASEY(27))	TIGP0150
BEG(1)=COMGET(IBASEX(26))	TIGP0151
BEG(2)=COMGET(IBASEY(26))	TIGP0152
DELX=(XEND-XBEG)/3360.	TIGP0153
DELY=(YEND-YBEG)/2400.	TIGP0154
C	TIGP0155
C FIND VIRTUAL SPACE TO SCREEN SPACE SCALING PARAMETERS	TIGP0156
C	TIGP0157
RDX2=1./(DELX*DELX)	TIGP0158
RDY2=1./(DELY*DELY)	TIGP0159
CALL SETWIN	TIGP0160
CALL GRID	TIGP0161
CALL LABEL(IBASEY(0))	TIGP0162
CALL LABEL(IBASEX(0))	TIGP0163
CALL DRAWIT(NL,NPTA,X,Y,ISUB)	TIGP0164
C	TIGP0165
C AXIS LABELS	TIGP0166
C	TIGP0167
CALL CHRSIZ(3)	TIGP0168
CALL TTITE(2320,3000,NTL,LABTL,80,0)	TIGP0169
CALL TTITE(2320,100,N2,LABX,80,0)	TIGP0170
CALL TTITE(450,1500,N1,LABY,80,1)	TIGP0171
C	TIGP0172
C MERGE HERE FOR INTERACTIVE FUNCTIONS (BELL)	TIGP0173
C	TIGP0174
200 IF(LCNT.LT.220) GO TO 530	TIGP0175
CALL CHRSIZ(4)	TIGP0176
IF(NPTOT.EQ.1) GO TO 240	
CALL GETVAL(ICHAR,X0,Y0)	
210 IF(ICHAR.LE.64.OR.ICHAR.GE.95)GO TO 220	
ICHAR=ICHAR-64	
ICHECK=IQUICK(ICHAR)	
IF(ICHECK.EQ.0) GO TO 220	
GO TO (300,400,440,200,460,500,540,560,590,455),ICHECK	
220 LCNT=LCNT-LDEL	
CALL NOTATE(0,LCNT,10,MSG4)	
GO TO 200	
C	
C ADD POINT AFTER OR BEFORE SPECIFIED POINT (A OR B)	
C	
C CHECK IF C COMMAND AND FIRST POINT.	
C	
230 IF(NPTOT.EQ.0)GO TO 460	TIGP0189
240 LCNT=LCNT-LDEL	TIGP0190
CALL NOTATE(0,LCNT,20,MSG1)	TIGP0191

250	CALL GETVAL(ICHAR,X0,Y0)	TIGP0192
C	CHECK FOR NEW LINE COMMAND	TIGP0193
	IF(ICHAR.EQ.86) GO TO 580	TIGP0194
C	CHECK FOR ADD AFTER	TIGP0195
260	IF(ICHAR.EQ.65) GO TO 270	TIGP0196
C	CHECK FOR MOVE	TIGP0197
	IF(ICHAR.EQ.77) GO TO 270	TIGP0198
C	IF NOT A B OR M GO TO NEW COMMAND	TIGP0199
	IF(ICHAR.NE.66) GO TO 210	TIGP0200
270	CALL POINTA(X0,Y0)	TIGP0201
	CALL MOVEA(X0,Y0)	TIGP0202
	IF(IOFF.EQ.0)CALL ANCHO(IS)	TIGP0203
	IF(ICHAR.EQ.77) GO TO 290	TIGP0204
	NPTOT=NPTOT+1	TIGP0205
C	INCREMENT STORAGE COUNTER	TIGP0206
	NSTOR=NSTOR+1	TIGP0207
	NPTA(ISAVE)=NPTA(ISAVE)+1	TIGP0208
C	MOVE POINTER OF CLOSEST POINT TO END	TIGP0209
	ISUB(NSTOR)=ISUB(JSAVE)	TIGP0210
C	CHANGE CLOSEST POINTER TO ACCESS LAST POINT	TIGP0211
	ISUB(JSAVE)=NSTOR	TIGP0212
	IF(ICHAR.EQ.65) GO TO 280	TIGP0213
C	MOVE OLD POINT TO LAST POINT ( INSERT BEFORE)	TIGP0214
	X(NSTOR)=X(JSAVE)	TIGP0215
	Y(NSTOR)=Y(JSAVE)	TIGP0216
	GO TO 290	TIGP0217
C	NEW POINT ADD AFTER	TIGP0218
280	IF(KLAST.EQ.JSAVE) KLAST=NSTOR	TIGP0219
	JSAVE=NSTOR	TIGP0220
290	X(JSAVE)=X0	TIGP0221
	Y(JSAVE)=Y0	TIGP0222
	GO TO 250	TIGP0223
C		TIGP0224
C	DELETE POINT (D)	TIGP0225
C		TIGP0226
300	DSAVE=1.E40	TIGP0227
	IF(NPTOT.EQ.0) GO TO 200	
	IS=64	TIGP0228
	NSUM=1	TIGP0229
	K=1	TIGP0230
	DO 340 I=1,NL	TIGP0231
	NEND=NSUM+NPTA(I)-1	TIGP0232
	DO 330 J=NSUM,NEND	TIGP0233
	IF(NLINE.EQ.0) GO TO 310	TIGP0234
	IF(NDRAW(I).EQ.0) GO TO 320	TIGP0235
310	XDX=X(K)-X0	TIGP0236
	YDY=Y(K)-Y0	TIGP0237
	DIST=XDX*XDX#RDX2+YDY*YDY#RDY2	TIGP0238

IF(DIST.GE.DSAVE)GO TO 320	TIGP0239
DSAVE=DIST	TIGP0240
JSAVE=K	TIGP0241
ISAVE=I	TIGP0242
320 KLAST=K	TIGP0243
330 K=ISUB(K)	TIGP0244
340 NSUM=NEND+1	TIGP0245
IS=ISAVE+64	TIGP0246
350 IF(IS.LE.90)GO TO 360	TIGP0247
IS=IS-90	TIGP0248
GO TO 350	TIGP0249
360 CALL POINTA(X(JSAVE),Y(JSAVE))	TIGP0250
IF(IOFF.EQ.0)CALL ANCHO(IS)	TIGP0251
IF(ICHAR.NE.4) GO TO 230	TIGP0252
NPTOT=NPTOT-1	TIGP0253
K=JSIZE	TIGP0254
C IF DELETED POINT IS LAST ONE SKIP SHIFT	TIGP0255
IF(KLAST.EQ.JSAVE)GO TO 370	TIGP0256
C GET POINTER OF NEXT POINT	TIGP0257
K=ISUB(JSIZE)	TIGP0258
C TRANSFER POINTER OF NEXT POINT TO DELETED POINT	TIGP0259
ISUB(JSIZE)=ISUB(K)	TIGP0260
C MOVE VALUE OF NEXT POINT TO DELETED POINT	TIGP0261
X(JSIZE)=X(K)	TIGP0262
Y(JSIZE)=Y(K)	TIGP0263
370 IF(NPTOT.EQ.1)NSTOR=1	TIGP0264
C ZERO DELETED POINTER	TIGP0265
ISUB(K)=0	TIGP0266
NPTA(ISAVE)=NPTA(ISAVE)-1	TIGP0267
IF(NPTA(ISAVE).GT.0)GO TO 390	TIGP0268
NPTA(ISAVE)=0	TIGP0269
J=0	TIGP0270
DO 380 I=1,NL	TIGP0271
IF(I.EQ.ISAVE)GO TO 380	TIGP0272
J=J+1	TIGP0273
NPTA(J)=NPTA(I)	TIGP0274
VLBL(J)=VLBL(I)	TIGP0275
380 CONTINUE	TIGP0276
NPTA(NZL)=0	TIGP0277
NL=NL-1	TIGP0278
390 GO TO 200	TIGP0280
C END (E)	TIGP0281
C	TIGP0282
400 CALL NEWPAG	TIGP0283
L=1	TIGP0284
DO 431 I=2,NPTOT	
K=ISUB(L)	

```

IF(I.EQ.K) GO TO 431
J=K
JLEFT=NPTOT+1-I
DO 420 KK=1,JLEFT
IF(J.EQ.I) GO TO 430
JO=J
420 J=ISUB(JO)
430 ISUB(JO)=K
ISUB(L)=I
IS=ISUB(I)
ISUB(I)=ISUB(K)
ISUB(K)=IS
XS=X(I)
X(I)=X(K)
X(K)=XS
XS=Y(I)
Y(I)=Y(K)
Y(K)=XS
431 L=I
RETURN
C
C   FORMAT (F)
C
440 LCNT=LCNT-LDEL
CALL MOVABS(0,LCNT)
CALL ANMODE
IY=(Y0-YBEG)/DELY+300
II=(3045-IY)/50+1
IF(II.LT.1) II=1
IF(II.GT.NL) II=NL
PRINT 450,MODE(II)
450 FORMAT(* ITIP = *,I2)
CALL GETIN(1,VTEM)
MODE(II)=VTEM(1)
LCNT=LCNT-LDEL
C
C   IF F OUTSIDE OF AXIS THE SET ALL CURVE MODES
C
IF(X0.LE.TXMIN) GO TO 200
DO 451 I=1,30
451 MODE(I)=VTEM(1)
GO TO 200
C
C   CHANGE GRID OPTION
C
455 IGRID=-IGRID
GO TO 200
C
TIGP0297
TIGP0298
TIGP0299
TIGP0304
TIGP0305
TIGP0306
TIGP0307
TIGP0308
TIGP0309
TIGP0310
0354
0355
0356
0357
TIGP0311
TIGP0312
TIGP0313
TIGP0314
TIGP0317
TIGP0318

```

C	NEW LINE (N)	TIGP0319
C		TIGP0320
460	NPTOT=NPTOT+1	TIGP0321
	NSTOR=NSTOR+1	TIGP0322
	IF(NPTOT.NE.1) ISUB(KLAST)=NSTOR	TIGP0323
	KLAST=NSTOR	TIGP0324
	IF(IPLOT+ITAB.EQ.0)GO TO 580	
461	X(NSTOR)=X0	TIGP0325
	Y(NSTOR)=Y0	TIGP0326
	NL=NL+1	TIGP0327
	IS=NL+64	TIGP0328
C	NO DATA THEN DO[T SYMBOL IT	
C	IF(IPLOT.EQ.0) GO TO 490	TIGP0329
470	IF(IS.LE.90)GO TO 480	TIGP0330
	IS=IS-90	TIGP0331
	GO TO 470	TIGP0332
480	CALL POINTA(X0,Y0)	TIGP0333
	IF(IOFF.EQ.0)CALL ANCHO(IS)	TIGP0334
490	NPTA(NL)=1	TIGP0335
	NL1=NL+1	TIGP0336
	NPTA(NL1)=0	TIGP0337
	LCNT=LCNT-LDEL	TIGP0338
	CALL NOTATE(0,LCNT,20,MSG2)	TIGP0339
	LCNT=LCNT-LDEL	TIGP0340
	CALL MOVABS(0,LCNT)	TIGP0341
	CALL ANMODE	TIGP0342
	CALL GETIN(1,VLBL(NL))	TIGP0343
	ISAVE=NL	TIGP0344
	JSAVE=NSTOR	TIGP0345
	IF(IPLOT.EQ.0) GO TO 530	
	GO TO 240	
C	PLOT (P)	TIGP0347
C		TIGP0348
C		TIGP0349
C	CHECK FOR TABLET MODE, SKIP SPECIAL P SECTION IF TABLET	
C		
500	IF(ITAB.EQ.1) GO TO 530	TIGP0350
	IF(X0.GT.TXMIN) GO TO 530	TIGP0351
	IF(NLINE.GT.0) GO TO 520	TIGP0352
	DO 510 I=1,NL	
510	NDRAW(I)=0	TIGP0353
520	IY=(Y0-YBEG)/DELY+300	TIGP0354
	II=(3045-IY)/50+1	TIGP0355
	IF(II.LT.1) II=1	TIGP0356
	IF(II.GT.NL) II=NL	TIGP0357

NDRAW(II)=1	TIGP0358
NLINE=1	TIGP0359
GO TO 200	TIGP0360
530 CALL NEWPAG	TIGP0361
GO TO 40	TIGP0362
C	TIGP0363
C RESTORE WINDOW (R)	TIGP0364
C	TIGP0365
540 IF(X0.GT.TXMIN.OR.NLINE.EQ.0) GO TO 550	TIGP0366
NLINE=0	TIGP0367
GO TO 200	TIGP0368
550 IWIN=0	TIGP0369
GO TO 530	TIGP0370
C	TIGP0371
C SHOW VALUE (S)	TIGP0372
C	TIGP0373
560 LCNT=LCNT-LDEL	TIGP0374
CALL MOVABS(0,LCNT)	TIGP0375
CALL ANMODE	TIGP0376
PRINT 570,X0,Y0	TIGP0377
570 FORMAT(*X=*,G13.5,/,*Y=*,G13.5)	TIGP0378
LCNT=LCNT-LDEL	TIGP0379
GO TO 200	TIGP0380
C	TIGP0381
C VALUE IN (V)	TIGP0382
C	TIGP0383
580 LCNT=LCNT-LDEL	TIGP0384
CALL NOTATE(0,LCNT,10,MSG5)	TIGP0385
LCNT=LCNT-LDEL	TIGP0386
CALL MOVABS(0,LCNT)	TIGP0387
CALL ANMODE	TIGP0388
CALL GETIN(2,VTEM)	TIGP0389
X0=VTEM(1)	TIGP0390
Y0=VTEM(2)	TIGP0391
C	
C CHECK FOR N COMMAND VALUE INPUT SECTION.	
C	
IF(IPLOT.EQ.0) GO TO 461	TIGP0392
LCNT=LCNT-LDEL	TIGP0393
CALL NOTATE(0,LCNT,13,MSG6)	TIGP0394
CALL TINPUT(ICHAR)	TIGP0395
GO TO 260	TIGP0396
C	TIGP0397
C WINDOW (W)	TIGP0398
C	
590 CALL GETVAL(ICHAT,X1,Y1)	TIGP0399
AXMIN=AMIN1(X0,X1)	TIGP0400
AXMAX=AMAX1(X0,X1)	TIGP0401

AYMIN=AMIN1(Y0,Y1)	TIGP0402
AYMAX=AMAX1(Y0,Y1)	TIGP0403
IWIN=1	TIGP0404
GO TO 530	TIGP0405
END	TIGP0406
CGETVAL	GETV0001
SUBROUTINE GETVAL(ICHAR,XV,YV)	GETV0002
COMMON/TEKGPPR/DUM(3),ICL,ITAB,ITABS,XS,YS,DUM2(20),NLINE	GETV0003
C ICL=0 INITIALIZE TABLET	GETV0004
C IC=0 NOT IN CONTINUOUS MODE	GETV0005
C ITAB=0 SCREEN CURSER	GETV0006
C****	GETV0007
C**** GET VALUE AND CHARACTER FROM CROSS HAIRS OR TABLET	GETV0008
C****	GETV0009
C****	GETV0010
C**** CHECK FOR TABLE INPUTS	GETV0011
C****	GETV0012
10 IF(ITAB.EQ.1)GO TO 20	GETV0013
CALL VCURSR(ICHAR,XV,YV)	GETV0014
C****	GETV0015
C**** CHECK TO SEE IF SCREEN COMMAND WAS TO ACTIVATE TABLET	GETV0016
C****	GETV0017
IF(ICHAR.NE.84)GO TO 30	GETV0018
ITAB=1	GETV0019
ICL=ITABS	GETV0020
C****	GETV0021
C**** SET TABLET LAST CHARACTER (ALSO USED AS A FLAG FOR INITIALIZATION	GETV0022
C****	GETV0023
20 ICHAR=ICL	GETV0024
CALL TABVU(ICHAR,XV,YV)	GETV0025
C****	GETV0026
C**** SAVE LAST TABLET CHARACTER COMMAND	GETV0027
C****	GETV0028
ICL=ICHAR	GETV0029
NLINE=0	GETV0030
C	
C SET FLAG TO PLOT ALL LINES IN TABLET MODE	
C	
C****	GETV0031
C**** CHECK FOR TABLET HALT COMMAND	GETV0032
C****	GETV0033
IF(ICL.NE.72)GO TO 30	GETV0034
C****	GETV0035
C**** TURN OFF TABLET AND SAVE LAST COMMAND	GETV0036
C****	GETV0037
ITAB=0	GETV0038
ITABS=ICL	GETV0039
RETURN	GETV0040

```

30 IF(ICHAR.EQ.69)ITABS=0 GETV0041
    RETURN GETV0042
    END GETV0043
CTABVU TABV0001
      SUBROUTINE TABVU(ICHAR,XV,YV)
      COMMON/TEKGPPR/LDEL,LCNT,MAXSR,LTV(5)
      1 ,LS,MX1,MY1,MX2,MY2,XB,YB,FACX,FACY,ANG,MXB,MYB TABV0002
      DIMENSION MSG1(54),MSG2(43),MSG3(43),MSG4(18),ICONV(2,10),
      1 IRETN(2,10),XITEM(2) TABV0003
      DATA ((ICONV(I,J),J=1,10),I=1,2)/65,66,67,68,69,71,72,
      1                               105,78,80,82,83,86,87, TABV0004
      1                               32,32,32,32,32,32/ TABV0005
      DATA ((IRETN(I,J),J=1,10),I=1,2)/ 0, 0, 0, 0, 1, 1, 1, TABV0006
      1                               1, 0, 1, 1, 0, 1, 0, TABV0007
      1                               1, 1, 1, 1, 1, 1, 1/ TABV0008
      DATA MSG1/ 83,113,117, 97,114,101, 32,109,101,110, TABV0009
      1                               117, 32,119,105,116,104, 32,116, 97, 98, TABV0010
      1                               108,101,116, 32, 97,110,100, 32,116,111, TABV0011
      1                               117, 99,104, 32,117,112,112,101,114, 32, TABV0012
      1                               108,101,102,116, 32,109,101,110,117, 32, TABV0013
      1                               100,111,116, 46/ TABV0014
      DATA MSG2/ 84,111,117, 99,104, 32, 97,120,105,115, TABV0015
      1                               32,111,114,105,103,105,110, 32, 97,110, TABV0016
      1                               100, 32,101,110,116,101,114, 32,118, 97, TABV0017
      1                               108,117,101,115, 32, 88, 32, 97,110,100, TABV0018
      1                               32, 89, 46/ TABV0019
      DATA MSG3/ 84,111,117, 99,104, 32, 32, 32, 97,120, TABV0020
      1                               105,115, 32, 97,116, 32,109, 97,120, 32, TABV0021
      1                               108,101,110,103,116,104, 32, 97,110,100, TABV0022
      1                               32,101,110,116,101,114, 32,118, 97,108, TABV0023
      1                               117,101, 46/ TABV0024
      DATA MSG4/ 76, 97,115,116, 32, 99,111,109,109, 97, TABV0025
      1                               110,100, 32,119, 97,115, 32, 32/ TABV0026
      IF(ICHAR.NE.0)GO TO 30 TABV0027
      C TABV0028
          TABLET HAS NOT BEEN SET CHECK IT TABV0029
          LS=100 TABV0030
          CALL TABINT(1,0,0) TABV0031
          CALL NEWPAG TABV0032
          LCNT=3120-LDEL TABV0033
      C GET MENU POSITION TABV0034
          CALL NOTATE(0,LCNT,54,MSG1) TABV0035
          CALL BELL TABV0036
          CALL ONEPNT(MX1,MY1) TABV0037
          MX2=MX1+1000 TABV0038
          MY2=MY1-200 TABV0039
          GO TO 20 TABV0040
      10 LCNT=3120 TABV0041
          CALL NEWPAG TABV0042
          TABV0043 TABV0044
          TABV0045

```

C	20 LCNT=LCNT-LDEL	TABV0046
	GET COORDINATE INTERSECTION	TABV0047
	CALL NOTATE(0,LCNT,43,MSG2)	TABV0048
	CALL BELL	TABV0049
	CALL ONEPNT(MXB,NYB)	TABV0050
	LCNT=LCNT-LDEL	TABV0051
	CALL MOVABS(0,LCNT)	TABV0052
	CALL ANMODE	TABV0053
	CALL GETIN(2,XTEM)	TABV0054
	XB=XTEM(1)	TABV0055
	YB=XTEM(2)	TABV0056
	LCNT=LCNT-LDEL	TABV0057
	MSG3(7)=88	TABV0058
C	GET X AXIS POSITION MAX	TABV0059
	CALL NOTATE(0,LCNT,43,MSG3)	TABV0060
	CALL BELL	TABV0061
	CALL ONEPNT(MXM,NXM)	TABV0062
	LCNT=LCNT-LDEL	TABV0063
	CALL MOVABS(0,LCNT)	TABV0064
	CALL ANMODE	TABV0065
C	GET VALUE AT POSITION	TABV0066
	CALL GETIN(1,XM)	TABV0067
	DX=MXM-MXB	TABV0068
	DY=NXM-NYB	TABV0069
C	COMPUTE ANGLE CORRECTION	TABV0070
	ANG=ATAN2(DY,DX)	TABV0071
	LCNT=LCNT-LDEL	TABV0072
	MSG3(7)=89	TABV0073
C	GET Y AXIS POSITION MAX	TABV0074
	CALL NOTATE(0,LCNT,43,MSG3)	TABV0075
	CALL BELL	TABV0076
	CALL ONEPNT(NYM,NYM)	TABV0077
	LCNT=LCNT-LDEL	TABV0078
	CALL MOVABS(0,LCNT)	TABV0079
	CALL ANMODE	TABV0080
C	GET VALUE AT POSITION	TABV0081
	CALL GETIN(1,YM)	TABV0082
	DY=NYM-NYB	TABV0083
	COSA=COS(ANG)	TABV0084
C	SET UP COMMON FACTORS FOR ANGLE CORRECTIONS	TABV0085
	FACX=(XM-XB)*COSA/DX	TABV0086
	FACY=(YM-YB)*COSA/DY	TABV0087
	INIT=1	TABV0088
	XV=XM	TABV0089
	YV=YM	TABV0090
	ICHAR=87	TABV0091
C	RETURN PLOT COMMAND	TABV0092
	RETURN	TABV0093

```

C      CHECK FOR TABLET INITIALIZED          TABV0094
30 IF(INIT.NE.1)GO TO 40                 TABV0095
      ICHAR=87                            TABV0096
      XV=XB                               TABV0097
      YV=YB                               TABV0098
      INIT=0                               TABV0099
      RETURN                               TABV0100
40 CALL BELL                           TABV0101
      CALL ONEPNT(IX,IY)                  TABV0102
C      CHECK TO SEE IF POINT SENT IS A MENU COMMAND    TABV0103
      IF(IX.GT.MX2.OR.IX.LT.MX1)GO TO 50    TABV0104
      IF(IY.GT.MY1.OR.IY.LT.MY2)GO TO 50    TABV0105
      IC=(IX-MX1)/LS+1                    TABV0106
      IR=(MY1-IY)/LS+1                   TABV0107
C      CONVERT ROW AND COLUMN POSITION TO COMMAND CHARACTER   TABV0108
      ICHAR=ICONV(IR,IC)                  TABV0109
      IF(ICHR.EQ.32) RETURN               TABV0110
      MSG4(18)=ICHAR                     TABV0111
      LCNT=LCNT-LDEL                     TABV0112
C      LAST MESSAGE COMMAND             TABV0113
      CALL NOTATE(0,LCNT,18,MSG4)        TABV0114
      IF(ICHR.EQ.105)GO TO 10            TABV0115
      IF(IRETN(IR,IC).EQ.1)RETURN       TABV0116
      GO TO 40                           TABV0117
C      CONVERT TABLET UNITS TO VIRTUAL UNITS WITH ANGLE CORRECTION   TABV0118
50 DX=IX-MXB                         TABV0119
      DY=IY-MYB                         TABV0120
      IF(DX.EQ.0.)DX=1.E-20              TABV0121
      R=SQRT(DX*DX+DY*DY)              TABV0122
      ANGR=ATAN2(DY,DX)-ANG           TABV0123
      XV=R*FACX*COS(ANGR)+XB         TABV0124
      YV=R*FACY*SIN(ANGR)+YB         TABV0125
      RETURN                             TABV0126
      END                                TABV0127
CDRAWIT
      SUBROUTINE DRAWIT(NL,NPTA,X,Y,ISUB)
      COMMON/TKTRNX/ITEKC(60)
      COMMON/TEKGPPR/DUM(20),EN(2),DEL(2),BEG(2),RDX2,RDY2,NLINE,
      1 NDRAW(30),MODE(30)
      DIMENSION QSY(306),QSX(306),NPTA(1),X(1),Y(1),ISUB(1)
      EQUIVALENCE (IOFF,ITEKC(30))          DRAW0001
C
C      0 SYMBOLS 1 LINE 2 SPLINE WRT X 3 SPLINE WRT Y 4 ARC FIT 5 CLOSED   DRAW0002
C
      IT=64                               DRAW0003
      K=1                                 DRAW0004
      NSUM=1                             DRAW0005
      DRAW0006
      DRAW0007
      DRAW0008
      DRAW0009
      DRAW0010
      DRAW0011
      DRAW0012
      DRAW0013
      DRAW0014
      DRAW0015

```

```

NC=0                                DRAW0016
DO 290 I=1,NL
ISYM=MCDE(I)
ITYP=IABS(ISYM)
IF(ITYP.GT.1) GO TO 40               DRAW0008
NEND=NSUM+NPTA(I)-1                 DRAW0017
IT=IT+1                             DRAW0018
IF(IT.GT.90)IT=65                   DRAW0019
DO 20 J=NSUM,NEND                  DRAW0020
IF(NLINE.EQ.0) GO TO 10             DRAW0021
IF(NDRAW(I).EQ.0) GO TO 20          DRAW0022
10 XP=X(K)                          DRAW0023
YP=Y(K)                            DRAW0024
IF(J.EQ.NSUM) CALL MOVEA(XP,YP)      DRAW0025
IF(ITYP.EQ.1) CALL DRAWA(XP,YP)      DRAW0026
IF(ISYM.LT.0) GO TO 20              DRAW0027
CALL MOVEA(XP,YP)                   DRAW0028
IF(IOFF.EQ.0) CALL ANCHO(IT)        DRAW0029
CALL MOVEA(XP,YP)
20 K=ISUB(K)                        DRAW0030
30 NSUM=NEND+1                      DRAW0031
GO TO 290                           DRAW0032
C
C   PLOT WITH SPLINE
C
40 NS=NC                            DRAW0033
NPT=NPTA(I)                         DRAW0034
NC=NC+NPT                           DRAW0035
IT=IT+1                            DRAW0036
IF(IT.GT.90)IT=65                   DRAW0042
IF(NLINE.EQ.0) GO TO 60             DRAW0043
IF(NDRAW(I).NE.0) GO TO 60          DRAW0044
C   LOCATE POINTER OF NEXT LINE
DO 50 L=1,NPT                      DRAW0045
50 K=ISUB(K)                        DRAW0046
GO TO 290                           DRAW0047
60 JFIT=2                           DRAW0048
YO=Y(K)                            DRAW0049
K1=ISUB(K)                          DRAW0050
IF(ITYP.GT.2) GO TO 80             DRAW0051
XO=X(K)                            DRAW0052
C   CHECK X DATA FOR ASCENDING ORDER
DO 70 L=2,NPT                      DRAW0053
X1=X(K1)                           DRAW0054
IF(X1.LE.XO) GO TO 110            DRAW0055
K1=ISUB(K1)                         DRAW0056
70 XO=X1                           DRAW0057
GO TO 210                           DRAW0058
                                         DRAW0059
                                         DRAW0060
                                         DRAW0061
                                         DRAW0062
                                         DRAW0063
                                         DRAW0064

```

```

80 IF(ITYP.GT.3) GO TO 100          DRAWC055
C   CHECK Y DATA FOR ASCENDING ORDER  DRAWC066
DO 90 L=2,NPT                      DRAWC067
Y1=Y(K1)                            DRAWCC68
IF(Y1.LE.Y0) GO TO 110              DRAWC069
K1=ISUB(K1)                         DRAW0070
90 Y0=Y1                            DRAWC071
GO TO 210                           DRAW0072
100 JFIT=ITYP-2                     DRAW0073
110 NCIR=0                          DRAW0074
IF(JFIT.EQ.3) NCIR=-NPT/2-1        DRAW0075
MPT=NPT-2*NCIR                     DRAW0076
QSY(1)=MPT                          DRAWC077
QSX(1)=MPT                          DRAW0078
S=0.                                DRAW0079
KA=NS                             DRAW0080
KO=KA                             DRAWC081
KE=KO+NPT                         DRAW0082
KSAVE=K                           DRAW0083
KA=KA+NCIR                        DRAW0084
DO 160 M=1,MPT                     DRAW0085
M1=M+1                            DRAW0086
KA=KA+1                           DRAW0087
IF(KA.GT.KO) GO TO 130            DRAW0088
NDO=NPT+NCIR                      DRAW0089
DO 120 II=1,NDO                   DRAW0090
120 K=ISUB(K)                      DRAW0091
KA=KA+NPT                         DRAW0092
GO TO 140                           DRAW0093
130 IF(KA.NE.(KE+1))GO TO 140    DRAW0094
JSAVE=K                           DRAW0095
K=KSAVE                           DRAWC096
KA=KA-NPT                         DRAW0097
140 CONTINUE                        DRAW0098
L=M1-MPT                          DRAW0099
YYYP=Y(K)                          DRAW0100
XXXP=X(K)                          DRAW0101
K=ISUB(K)                          DRAW0102
IF(M.EQ.1) GO TO 150              DRAW0103
DS=SQRT(RDX2*(XXXP-X0)**2+RDY2*(YYYP-Y0)**2)
S=S+DS
150 X0=XXXP                         DRAW0104
Y0=YYYP                           DRAW0105
QSX(M1)=S                          DRAW0106
QSY(M1)=S                          DRAW0107
QSX(L)=XXXP                        DRAW0108
160 QSY(L)=YYYP                     DRAW0109
KA=KO+NPT                         DRAW0110
                                         DRAW0111
                                         DRAW0112

```

QSX(L+1)=0.	DRAW0113
QSY(L+1)=0.	DRAW0114
QSX(L+2)=1.	DRAW0115
QSY(L+2)=1.	DRAW0116
XO=QSX(MPT+2-NCIR)	DRAW0117
YO=QSY(MPT+2-NCIR)	DRAW0118
CALL MOVEA(XO, YO)	DRAW0119
IF(IOFF.EQ.0) CALL ANCHO(IT)	DRAW0120
CALL MOVEA(XO, YO)	DRAW0121
SCK=QSX(3-NCIR)	DRAW0122
S=QSX(2-NCIR)	DRAW0123
IF(NPT.LE.1) GO TO 290	DRAW0124
DC=40.	DRAW0125
DS=40.	DRAW0126
NCK=2	DRAW0127
170 S=S+DS	DRAW0128
XP=SPLNQ1(1, QSX, S)	DRAW0129
YP=SPLNQ1(1, QSY, S)	DRAW0130
DCK=SQRT(RDX2*(XO-XP)**2+RDY2*(YO-YP)**2)	DRAW0131
DS= DC*DS/DCK	DRAW0132
180 IF(S.LT.SCK) GO TO 200	DRAW0133
NSYM=MPT+1+NCK-NCIR	DRAW0134
XS=QSX(NSYM)	DRAW0135
YS=QSY(NSYM)	DRAW0136
CALL DRAWA(XS, YS)	DRAW0137
IF(ISYM.LE.0.AND.NCK.NE.NPT) GO TO 190	DRAW0138
CALL MOVEA(XS, YS)	DRAW0139
IF(IOFF.EQ.0) CALL ANCHO(IT)	DRAW0140
CALL MOVEA(XS, YS)	DRAW0141
190 NCK=NCK+1	DRAW0142
SCK=QSX(NCK+1-NCIR)	DRAW0143
IF(NCK.LE.NPT+JFIT-2) GO TO 180	DRAW0144
IF(JFIT.EQ.3) K <sub>2</sub> JSAVE	DRAW0145
GO TO 290	DRAW0146
200 CALL DRAWA(XP, YP)	DRAW0147
XO=XP	DRAW0148
YO=YP	DRAW0149
GO TO 170	DRAW0150
210 QSX(1)=MPT	DRAW0151
DO 240 M=1, NPT	DRAW0152
M=M+1	DRAW0153
K=M-NPT	DRAW0154
XP=X(K)	DRAW0155
YP=Y(K)	DRAW0156
IF(M.NE.1.AND.M.NE.NPT.AND.ISYM.LE.0) GO TO 220	DRAW0157
CALL MOVEA(XP, YP)	DRAW0158
IF(IOFF.EQ.0) CALL ANCHO(IT)	DRAW0159
	DRAW0160

220 IF(ITYP.NE.3) GO TO 230	DRAW0161
QSX(N)=YP	DRAW0162
QSX(L)=XP	DRAW0163
GO TO 240	DRAW0164
230 QSX(N)=XP	DRAW0165
QSX(L)=YP	DRAW0166
240 K=ISUB(K)	DRAW0167
QSX(L+1)=0.	DRAW0168
QSX(L+2)=1.	DRAW0169
XEN=QSX(NPT+1)	DRAW0170
XIN=QSX(2)	DRAW0171
IF ITP=ITYP-1	DRAW0172
BCK=BEG(IFITP)	DRAW0173
ECK=EN(IFITP)	DRAW0174
DELT=DEL(IFITP)*30.	DRAW0175
IF(XIN.LT.BCK) XIN=BCK	DRAW02176
IF(XEN.GT.ECK) XEN=ECK	DRAW0177
KILL=0	DRAW0178
DO 280 M=1,200	DRAW0179
XI=XIN+DELT*(M-1)	DRAW0180
IF(XI.LT.XEN) GO TO 250	DRAW0181
KILL=1	DRAW0182
XI=XEN	DRAW0183
250 YI=SPLNQ1(1,QSX,XI)	DRAW0184
IF(ITYP.EQ.3) GO TO 260	DRAW0185
XP=XI	DRAW0186
YP=YI	DRAW0187
GO TO 270	DRAW0188
260 XP=YI	DRAW0189
YP=XI	DRAW0190
270 IF(M.EQ.1) CALL MOVEA(XP,YP)	DRAW0191
CALL DRAWA(XP,YP)	DRAW0192
IF(NPT.EQ.1) GO TO 290	DRAW0193
IF(KILL.EQ.1) GO TO 290	DRAW0194
280 CONTINUE	DRAW0195
290 CONTINUE	DRAW0196
300 RETURN	DRAW0197
END	DRAW0198
CSPLNQ1	SPLN0001
FUNCTION SPLNQ1 (NLOC,X,XINDEP)	SPLN0002
C*** LOCAL CUBIC FIT 8/9/77 M.J. CADDY	SPLN0003
DIMENSION X(1),QM(3)	SPLN0004
EQUIVALENCE (QM(1),T3),(QM(2),Q2),(QM(3),Q3)	SPLN0005
XIN=XINDEP	SPLN0006
NS=NLOC	SPLN0007
NOPTS=X(NS)	SPLN0008
ID=NS-NOPTS	SPLN0009
NSP1=NS+1	

```

NSP2=NS+2 SPLN0010
IF(NOPTS.LE.1) GO TO 130 SPLN0011
IF(NOPTS.GT.2) GO TO 10 SPLN0012
N=ID+NOPTS SPLN0013
T3=(X(N)-X(N-1))/(X(ID)-X(ID-1)) SPLN0014
M=ID SPLN0015
NTRAP=1 SPLN0016
GO TO 280 SPLN0017
10 NS2=NOPTS*2+NSP1 SPLN0018
L=X(NS2)
LSC=NS2+1 SPLN0019
IQMODE=X(LSC)
K=L+NS SPLN0020
NL=NSP1 SPLN0021
NH=ID SPLN0022
NTRAP=-1 SPLN0023
C*** BINARY SEARCH FOR INTERVAL SPLN0024
IF(XIN-X(ID))30,140,20 SPLN0025
20 NTRAP=0 SPLN0026
GO TO 150 SPLN0027
30 IF(XIN-X(NSP1))40,40,60 SPLN0028
40 NTRAP=1 SPLN0029
50 K=NSP2 SPLN0030
GO TO 160 SPLN0031
60 IF(L)120,120,70 SPLN0032
70 IF(XIN-X(K))80,100,100 SPLN0033
80 NH=K SPLN0034
K=K-1 SPLN0035
90 IF(XIN-X(K))110,100,100 SPLN0036
100 NL=K SPLN0037
GO TO 120 SPLN0038
110 NH=K SPLN0039
120 K=(NH-NL)/2+NL SPLN0040
IF(K-NL)90,140,90 SPLN0041
130 YOUT=X(NSP2)
GO TO 320 SPLN0042
140 LFAST=L-NH+NS SPLN0043
X(NS2)=NH-NS SPLN0044
150 K=NH SPLN0045
160 M=K SPLN0046
M=M+NOPTS SPLN0047
Y3=X(N-1)
X3=X(M-1)
C*** CHECK FOR FAST MODE AND EXTRAPOLATION SPLN0048
IF(NTRAP.GE.0) GO TO 180 SPLN0049
IF(IQMODE*L.EQ.0.OR.LFAST.NE.0) GO TO 180 SPLN0050
DO 170 I=1,3 SPLN0051
170 QM(I)=X(LSC+I) SPLN0052
SPLN0053
SPLN0054
SPLN0055
SPLN0056
SPLN0057

```

NADC-83030-80

GO TO 310	SPLNC058
180 Y4=X(N)	SPLNC059
X4=X(M)	SPLNC060
A3=X4-X3	SPLNC061
S3=(Y4-Y3)/A3	SPLNC062
IF(M.EQ.NSP2) GO TO 190	SPLNC063
X2=X(M-2)	SPLNC064
Y2=X(N-2)	SPLN0065
S2=(Y3-Y2)/(X3-X2)	SPLN0066
IF(M.EQ.ID) GO TO 200	SPLN0067
190 X5=X(M+1)	SPLN0068
Y5=X(N+1)	SPLN0069
S4=(Y5-Y4)/(X5-X4)	SPLNC070
IF(M.EQ.NSP2) S2=S3+S3-S4	SPLN0071
GO TO 210	SPLN0072
200 S4=S3+S3-S2	SPLN0073
210 IF(M.LE.(NSP2+1)) GO TO 220	SPLN0074
S1=(Y2-X(N-3))/(X2-X(M-3))	SPLN0075
GO TO 230	SPLN0076
220 S1=S2+S2-S3	SPLN0077
230 IF(M.GE.(ID-1)) GO TO 240	SPLN0078
S5=(X(N+2)-Y5)/(X(M+2)-X5)	SPLN0079
GO TO 250	SPLN0080
240 S5=S4+S4-S3	SPLN0081
250 W2=ABS(S4-S3)	SPLN0082
W3=ABS(S2-S1)	SPLN0083
SW=W2+W3	SPLN0084
IF(SW.NE.0.0) GO TO 260	SPLN0085
W2=0.5	SPLN0086
W3=0.5	SPLN0087
SW=1.0	SPLN0088
260 T3=(W2*S2+W3*S3)/SW	SPLN0089
W3=ABS(S5-S4)	SPLN0090
W4=ABS(S3-S2)	SPLN0091
SW=W3+W4	SPLN0092
IF(SW.NE.0.0) GO TO 270	SPLN0093
W3=0.5	SPLN0094
W4=0.5	SPLN0095
SW=1.0	SPLN0096
270 T4=(W3*S3+W4*S4)/SW	SPLN0097
IF(NTRAP.LT.0) GO TO 290	SPLN0098
IF(NTRAP.EQ.0) T3=T4	SPLN0099
280 IX=M-NTRAP	SPLN0100
C*** FAST EXIT FOR 2 POINTS AND LINEAR EXTRAPOLATION	SPLN0101
YOUT=X(IX+NOPTS)+(XIN-X(IX))*T3	SPLN0102
GO TO 320	SPLN0103
290 Q2=(2.0*(S3-T3)+S3-T4)/A3	SPLN0104
Q3=(-S3-S3+T3+T4)/(A3*A3)	SPLN0105

```

IF(IQMODE#LFAST.EQ.0) GO TO 310          SPLN0106
DO 300 I=1,3                            SPLN0107
300 X(LSC+I)=QM(I)                      SPLN0108
310 DX=XIN-X3                          SPLN0109
      YOUT=Y3+DX*(T3+DX*(Q2+DX*Q3))    SPLN0110
320 SPLNQ1=YOUT                         SPLN0111
      RETURN                               SPLN0112
      END                                  SPLN0113
CTTITE                                     TTIIT0001
      SUBROUTINE TTIITE(IX,IY,NTL,LABTL,NM,IA)   TTIIT0002
      DIMENSION LABTL(1),IP(136)                 TTIIT0003
C       NTL =NUMBER OF 10 CHARACTER WORDS        TTIIT0004
C       NM MAX CHARACTERS PER LINE            TTIIT0005
C       IA SWITCH, IA=0 HORIZ, IA=1 VERTICAL    TTIIT0006
C       IX SCREEN CENTER                      TTIIT0007
C       IY SCREEN CENTER                      TTIIT0008
C       IF(NTL.LE.0) RETURN                    TTIIT0009
NC=10*NTL                                 TTIIT0010
C       GET CHARACTER SIZE                  TTIIT0011
CALL CSIZE(IHORZ,IVERT)                   TTIIT0012
C       CONVERT LABEL TO ADE                TTIIT0013
CALL KAM2AS(NC,LABTL,IP)                  TTIIT0014
IX1=IX                                     TTIIT0015
IY1=IY                                     TTIIT0016
ITL1=0                                      TTIIT0017
NBLK=0                                      TTIIT0018
DO 70 K=1,NC                                TTIIT0019
      CHECK FOR LEADING BLANKS             TTIIT0020
      IF(IP(K).NE.32) GO TO 10           TTIIT0021
      IF(ITL1.EQ.0) GO TO 70             TTIIT0022
      NBLK=NBLK+1                      TTIIT0023
C       CHECK FOR 3 BLANKS TO TERMINATE LINE   TTIIT0024
      IF(NBLK.NE.3) GO TO 20             TTIIT0025
      ITL1=ITL1-2                      TTIIT0026
      GO TO 50                           TTIIT0027
10   NBLK=0                                    TTIIT0028
C       CHECK FOR MAX LINE LENGTH EXCEEDED   TTIIT0029
20   IF(ITL1.LT.NM) GO TO 30               TTIIT0030
      IF(IP(K).EQ.32) GO TO 50           TTIIT0031
30   ITL1=ITL1+1                          TTIIT0032
      IP(ITL1)=IP(K)                   TTIIT0033
      IF(K.LT.NC) GO TO 70             TTIIT0034
40   ITL1=ITL1-NBLK                      TTIIT0035
C       CHECK FOR VERTICAL OR HORIZ LABEL    TTIIT0036
50   IF(IA.NE.0) GO TO 60               TTIIT0037
      IX1=IX-IHORZ*ITL1*.5              TTIIT0038
      CALL NOTATE(IX1,IY1,ITL1,IP)       TTIIT0039
      IY1=IY1-IVERT*.1                 TTIIT0040

```

ITL1=0	TTIT0041
GO TO 70	TTIT0042
60 IY1=IY+IVERT*ITL1*.5	TTIT0043
CALL MOVABS(IX1,IY1)	TTIT0044
CALL VLABEL(ITL1,IP)	TTIT0045
IX1=IX1+IHORZ*.1.1	TTIT0046
ITL1=0	TTIT0047
70 CONTINUE	TTIT0048
END	TTIT0049
CGETIN	GETI0001
SUBROUTINE GETIN(NIN,Y)	GETI0002
C	GETI0003
C	GETI0004
C MICHAEL CADDY 3/19/78	GETI0005
DIMENSION Y(1),IC(80)	GETI0006
C FREE FORM INPUT CODE	GETI0007
NW=0	GETI0008
10 J=0	GETI0009
READ 20,IC	GETI0010
C CHECK FOR END OF FILE	GETI0011
IF.EOF(5).EQ.0)GO TO 30	GETI0012
NIN=NW	GETI0013
RETURN	GETI0014
20 FORMAT(80R1)	GETI0015
30 JC=0	GETI0016
JD=0	GETI0017
JS=1	GETI0018
NC=0	GETI0019
X=0.	GETI0020
40 J=J+1	GETI0021
C ONLY ONE CARD PER INPUT READ	GETI0022
C MODIFIED TO READ MORE THAN ONE CARD 4/26/78 MJC	
IF(J.GT.80) GO TO 10	GETI0023
I=IC(J)	GETI0024
C CHECK FOR VALID NUMERIC FIELD	GETI0025
IF(I.GT.32B.AND.I.LT.45B) GO TO 110	GETI0026
C IGNORE LEAD + SIGN	GETI0027
IF(I.EQ.45B) GO TO 40	GETI0028
C SET FLAG FOR NEGATIVE VALUE	GETI0029
IF(I.NE.46B) GO TO 50	GETI0030
JS=1	GETI0031
GO TO 40	GETI0032
C CHECK FOR DECIMAL	GETI0033
50 IF(I.NE.57B) GO TO 60	GETI0034
IF(JC.EQ.-1) GO TO 120	GETI0035
C IF THIS IS SECOND DECIMAL BLOW OFF TO ERROR CODE	GETI0036
JC=-1	GETI0037
GO TO 40	GETI0038

NADC-83030-60

C	CHARACTER IS BLANK TREAT AS COMMA IF NOT LEADING	GETI0039
60	IF(I.EQ.55B)GO TO 80	GETI0040
	IF(I.EQ.56B)GO TO 70	GETI0041
	GO TO 120	GETI0042
70	IF(NC.GT.0)GO TO 90	GETI0043
	NW=NW+1	GETI0044
	IF(NW.GT.NIN) RETURN	GETI0045
	GO TO 40	GETI0046
C	TWO COMMAS ..IGNORE THIS DATA FIELD AND GO ON TO NEXT	GETI0047
80	IF(NC.EQ.0) GO TO 40	GETI0048
C	SHIFT DECIMAL TO NUMBER	GETI0049
90	X=JS*X*10.**JD	GETI0050
	NW=NW+1	GETI0051
	Y(NW)=X	GETI0052
	IF(NW.GE.NIN) RETURN	GETI0053
	GO TO 30	GETI0054
110	JD=JD+JC	GETI0055
	NC=NC+1	GETI0056
C	ADD DIGIT TO NUMBER , , CAREFULLY	GETI0057
	X=X*10+(I-33B)	GETI0058
	GO TO 40	GETI0059
C	ERROR CODE	GETI0060
120	DO 130 K=1,80	GETI0061
130	IC(K)=55B	GETI0062
	IC(J)=47B	GETI0063
	PRINT 140 ,IC	GETI0064
140	FORMAT(2X,80R1)	GETI0065
	PRINT 150	GETI0066
150	FORMAT(* BAD FIELD, RE-ENTER DATA*)	GETI0067
	GO TO 10	GETI0068
	END	GETI0069

**DISTRIBUTION LIST**

**REPORT NO. NADC-83030-80**

**AIRTASK NO. A5365360/001F/B0598W0598002**

**Work Unit No. RF908**

**AIRTASK A2695100/001C/BW04770003-01**

**Work Unit No. XN401/X5701**

	<u>No. of Copies</u>
<b>NAVAIR.....</b> (2 for 526Z1A Donald Chestnutt) (2 for 53012D2 Terence Martin) (1 for A5360C James Byers)	5
<b>DDC .....</b>	12

DATE  
ILMED  
8